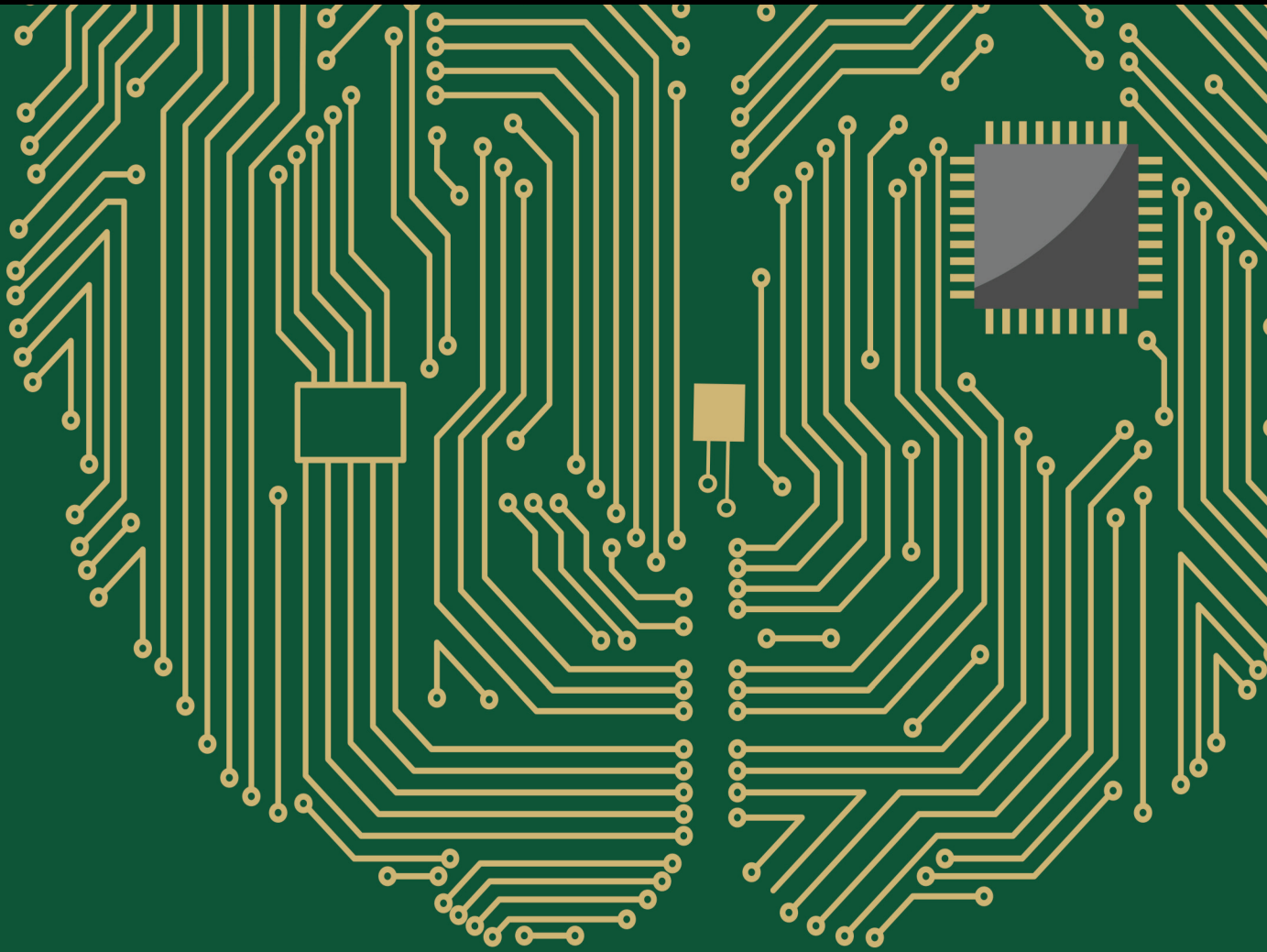


# Advanced Deep Learning and Neuro-Evolution Metaheuristic Techniques in Medical Applications

Lead Guest Editor: Mohamed Abd Elaziz

Guest Editors: Ahmed A. Ewees, Mohammed A. A. Al-qaness, Ali Ibrahim Rehab, Dalia Yousri, and Laith Abugaligah





---

**Advanced Deep Learning and Neuro-Evolution  
Metaheuristic Techniques in Medical  
Applications**

**Advanced Deep Learning and Neuro-  
Evolution Metaheuristic Techniques in  
Medical Applications**

Lead Guest Editor: Mohamed Abd Elaziz

Guest Editors: Ahmed A. Ewees, Mohammed A.  
A. Al-qaness, Ali Ibrahim Rehab, Dalia Yousri, and  
Laith Abugaligah



---


Copyright © 2023 Hindawi Limited. All rights reserved.

This is a special issue published in "Computational Intelligence and Neuroscience." All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

# Chief Editor

Andrzej Cichocki, Poland

## Associate Editors

Arnaud Delorme, France  
Cheng-Jian Lin , Taiwan  
Saeid Sanei, United Kingdom

## Academic Editors

Mohamed Abd Elaziz , Egypt  
Tariq Ahanger , Saudi Arabia  
Muhammad Ahmad, Pakistan  
Ricardo Aler , Spain  
Nouman Ali, Pakistan  
Pietro Aricò , Italy  
Lerina Aversano , Italy  
Ümit Ağbulut , Turkey  
Najib Ben Aoun , Saudi Arabia  
Surbhi Bhatia , Saudi Arabia  
Daniele Bibbo , Italy  
Vince D. Calhoun , USA  
Francesco Camastra, Italy  
Zhicheng Cao, China  
Hubert Cecotti , USA  
Jyotir Moy Chatterjee , Nepal  
Rupesh Chikara, USA  
Marta Cimitile, Italy  
Silvia Conforto , Italy  
Paolo Crippa , Italy  
Christian W. Dawson, United Kingdom  
Carmen De Maio , Italy  
Thomas DeMarse , USA  
Maria Jose Del Jesus, Spain  
Arnaud Delorme , France  
Anastasios D. Doulamis, Greece  
António Dourado , Portugal  
Sheng Du , China  
Said El Kafhali , Morocco  
Mohammad Reza Feizi Derakhshi , Iran  
Quanxi Feng, China  
Zhong-kai Feng, China  
Steven L. Fernandes, USA  
Agostino Forestiero , Italy  
Piotr Franaszczuk , USA  
Thippa Reddy Gadekallu , India  
Paolo Gastaldo , Italy  
Samanwoy Ghosh-Dastidar, USA

Manuel Graña , Spain  
Alberto Guillén , Spain  
Gaurav Gupta, India  
Rodolfo E. Haber , Spain  
Usman Habib , Pakistan  
Anandakumar Haldorai , India  
José Alfredo Hernández-Pérez , Mexico  
Luis Javier Herrera , Spain  
Alexander Hošovský , Slovakia  
Etienne Hugues, USA  
Nadeem Iqbal , Pakistan  
Sajad Jafari, Iran  
Abdul Rehman Javed , Pakistan  
Jing Jin , China  
Li Jin, United Kingdom  
Kanak Kalita, India  
Ryotaro Kamimura , Japan  
Pasi A. Karjalainen , Finland  
Anitha Karthikeyan, Saint Vincent and the  
Grenadines  
Elpida Keravnou , Cyprus  
Asif Irshad Khan , Saudi Arabia  
Muhammad Adnan Khan , Republic of  
Korea  
Abbas Khosravi, Australia  
Tai-hoon Kim, Republic of Korea  
Li-Wei Ko , Taiwan  
Raşit Köker , Turkey  
Deepika Koundal , India  
Sunil Kumar , India  
Fabio La Foresta, Italy  
Kuruva Lakshmana , India  
Maciej Lawrynczuk , Poland  
Jianli Liu , China  
Giosuè Lo Bosco , Italy  
Andrea Loddo , Italy  
Kezhi Mao, Singapore  
Paolo Massobrio , Italy  
Gerard McKee, Nigeria  
Mohit Mittal , France  
Paulo Moura Oliveira , Portugal  
Debajyoti Mukhopadhyay , India  
Xin Ning , China  
Nasimul Noman , Australia  
Fivos Panetsos , Spain









Evgeniya Pankratova , Russia  
Rocío Pérez de Prado , Spain  
Francesco Pistolesi , Italy  
Alessandro Sebastian Podda , Italy  
David M Powers, Australia  
Radu-Emil Precup, Romania  
Lorenzo Putzu, Italy  
S P Raja, India  
Dr.Anand Singh Rajawat , India  
Simone Ranaldi , Italy  
Upaka Rathnayake, Sri Lanka  
Navid Razmjooy, Iran  
Carlo Ricciardi, Italy  
Jatinderkumar R. Saini , India  
Sandhya Samarasinghe , New Zealand  
Friedhelm Schwenker, Germany  
Mijanur Rahaman Seikh, India  
Tapan Senapati , China  
Mohammed Shuaib , Malaysia  
Kamran Siddique , USA  
Gaurav Singal, India  
Akansha Singh , India  
Chiranjibi Sitaula , Australia  
Neelakandan Subramani, India  
Le Sun, China  
Rawia Tahrir , Iraq  
Binhua Tang , China  
Carlos M. Travieso-González , Spain  
Vinh Truong Hoang , Vietnam  
Fath U Min Ullah , Republic of Korea  
Pablo Varona , Spain  
Roberto A. Vazquez , Mexico  
Mario Versaci, Italy  
Gennaro Vessio , Italy  
Ivan Volosyak , Germany  
Leyi Wei , China  
Jianghui Wen, China  
Lingwei Xu , China  
Cornelio Yáñez-Márquez, Mexico  
Zaher Mundher Yaseen, Iraq  
Yugen Yi , China  
Qiangqiang Yuan , China  
Miaolei Zhou , China  
Michal Zochowski, USA  
Rodolfo Zunino, Italy

# Contents






## **Employing Atrous Pyramid Convolutional Deep Learning Approach for Detection to Diagnose Breast Cancer Tumors**

Ehsan Sadeghi Pour, Mahdi Esmaeili , and Morteza Romoozi  
Research Article (17 pages), Article ID 7201479, Volume 2023 (2023)






## **Classification of Breast Cancer Histopathological Images Using DenseNet and Transfer Learning**

Musa Adamu Wakili , Harisu Abdullahi Shehu , Md. Haidar Sharif , Md. Haris Uddin Sharif ,  
Abubakar Umar , Huseyin Kusetogullari , Ibrahim Furkan Ince , and Sahin Uyaver   
Research Article (31 pages), Article ID 8904768, Volume 2022 (2022)

## **Tuberculosis Detection in Chest Radiographs Using Spotted Hyena Algorithm Optimized Deep and Handcrafted Features**

Seifedine Kadry , Gautam Srivastava , Venkatesan Rajinikanth , Seungmin Rho , and Yongsung Kim   
Research Article (12 pages), Article ID 9263379, Volume 2022 (2022)




## **Optimizing Deep Learning Model for Software Cost Estimation Using Hybrid Meta-Heuristic Algorithmic Approach**

Ch Anwar ul Hassan , Muhammad Sufyan Khan, Rizwana Irfan, Jawaid Iqbal, Saddam Hussain , Syed Sajid Ullah , Roobaea Alroobaea , and Fazlullah Umar   
Research Article (20 pages), Article ID 3145956, Volume 2022 (2022)






## **A Review of the Role and Challenges of Big Data in Healthcare Informatics and Analytics**

Banan Jamil Awrahman , Chia Aziz Fatah, and Mzhda Yasin Hamaamin  
Review Article (10 pages), Article ID 5317760, Volume 2022 (2022)




## **An Improved Image Classification Method for Cervical Precancerous Lesions Based on ShuffleNet**

Shan Fang , Jiahui Yang, Minghui Wang, Chunhui Liu , and Shuang Liu   
Research Article (8 pages), Article ID 9675628, Volume 2022 (2022)





## **Arrhythmia Classification Algorithm Based on a Two-Dimensional Image and Modified EfficientNet**

Cui-fang Zhao , Wan-yun Yao , Mei-juan Yi , Chao Wan , and Yong-le Tian   
Research Article (10 pages), Article ID 8683855, Volume 2022 (2022)

## **A Comparison on LSTM Deep Learning Method and Random Walk Model Used on Financial and Medical Applications: An Example in COVID-19 Development Prediction**

Yifan Yao , Xinxin Li , and Qing Li   
Research Article (16 pages), Article ID 4383245, Volume 2022 (2022)

## **Novel Crow Swarm Optimization Algorithm and Selection Approach for Optimal Deep Learning COVID-19 Diagnostic Model**






Mazin Abed Mohammed , Belal Al-Khateeb, Mohammed Yousif, Salama A. Mostafa , Seifedine Kadry , Karrar Hameed Abdulkareem , and Begonya Garcia-Zapirain  
Research Article (22 pages), Article ID 1307944, Volume 2022 (2022)

**DCNN-FuzzyWOA: Artificial Intelligence Solution for Automatic Detection of COVID-19 Using X-Ray Images**

Abbas Saffari , Mohammad Khishe , Mokhtar Mohammadi , Adil Hussein Mohammed , and Shima Rashidi 


Research Article (11 pages), Article ID 5677961, Volume 2022 (2022)

**GC-CNNnet: Diagnosis of Alzheimer's Disease with PET Images Using Genetic and Convolutional Neural Network**

Morteza Amini , Mir Mohsen Pedram , AliReza Moradi , Mahdiah Jamshidi , and Mahshad Ouchani 

Research Article (15 pages), Article ID 7413081, Volume 2022 (2022)

**Weighted Joint Sentiment-Topic Model for Sentiment Analysis Compared to ALGA: Adaptive Lexicon Learning Using Genetic Algorithm**

Amjad Osmani  and Jamshid Bagherzadeh Mohasefi





Research Article (35 pages), Article ID 7612276, Volume 2022 (2022)

**Entropy and Gaussian Filter-Based Adaptive Active Contour for Segmentation of Skin Lesions**

Saleem Mustafa, Muhammad Waseem Iqbal , Toqir A. Rana , Arfan Jaffar, Muhammad Shiraz, Muhammad Arif, and Samia Allaoua Chelloug 



Research Article (10 pages), Article ID 4348235, Volume 2022 (2022)

**Predicting Divorce Prospect Using Ensemble Learning: Support Vector Machine, Linear Model, and Neural Network**

Mian Muhammad Sadiq Fareed , Ali Raza, Na Zhao , Aqil Tariq , Faizan Younas, Gulnaz Ahmed, Saleem Ullah, Syeda Fizzah Jillani , Irfan Abbas, and Muhammad Aslam


Research Article (15 pages), Article ID 3687598, Volume 2022 (2022)

**Multiclass Cancer Prediction Based on Copy Number Variation Using Deep Learning**

Haleema Attique, Sajid Shah , Saima Jabeen, Fiaz Gul Khan, Ahmad Khan, and Mohammed ELAffendi 


Research Article (11 pages), Article ID 4742986, Volume 2022 (2022)

**An Analysis of New Feature Extraction Methods Based on Machine Learning Methods for Classification Radiological Images**

Firoozeh Abolhasani Zadeh , Mohammadreza Vazifeh Ardalani, Ali Rezaei Salehi, Roza Jalali Farahani, Mandana Hashemi, and Adil Hussein Mohammed

Research Article (13 pages), Article ID 3035426, Volume 2022 (2022)

**5G Converged Network Resource Allocation Strategy Based on Reinforcement Learning in Edge Cloud Computing Environment**

Xuezhu Li 

Research Article (8 pages), Article ID 6174708, Volume 2022 (2022)



# Contents

## **Image Semantic Segmentation Method Based on Deep Fusion Network and Conditional Random Field**

Shuo Wang  and Yi Yang 


Research Article (9 pages), Article ID 8961456, Volume 2022 (2022)

## **WTD-PSD: Presentation of Novel Feature Extraction Method Based on Discrete Wavelet Transformation and Time-Dependent Power Spectrum Descriptors for Diagnosis of Alzheimer's Disease**

Ali Taghavarashidizadeh , Fatemeh Sharifi, Seyed Amir Vahabi, Aslan Hejazi, Mehrnaz SaghabTorbati, and Amin Salih Mohammed



Research Article (13 pages), Article ID 9554768, Volume 2022 (2022)

## **PSOWNNs-CNN: A Computational Radiology for Breast Cancer Diagnosis Improvement Based on Image Processing Using Machine Learning Methods**

Ashkan Nomani, Yasaman Ansari, Mohammad Hossein Nasirpour, Armin Masoumian, Ehsan Sadeghi Pour, and Amin Valizadeh 

Research Article (17 pages), Article ID 5667264, Volume 2022 (2022)

## **FDCNet: Presentation of the Fuzzy CNN and Fractal Feature Extraction for Detection and Classification of Tumors**

Sepideh Molaei, Niloofer Ghorbani , Fatemeh Dashtiahangar, Mohammad Peivandi , Yaghoob Poursad , and Mona Esmaili

Research Article (16 pages), Article ID 7543429, Volume 2022 (2022)

## **The Method for Identifying Employees' Emotions in Adverse States Incorporating PSO-kNN Algorithm and Multiple Physiological Parameters**

Jiaonan Han 





Research Article (10 pages), Article ID 4371162, Volume 2022 (2022)

## **Personalized Liver Cancer Risk Prediction Using Big Data Analytics Techniques with Image Processing Segmentation**

Anurag Jain , Ahmed Nadeem, Huda Majdi Altoukhi , Sajjad Shaukat Jamal, Henry kwame Atiglah , and Haitham Elwahsh

Research Article (11 pages), Article ID 8154523, Volume 2022 (2022)

## **Biosensor-Assisted Method for Abdominal Syndrome Classification Using Machine Learning Algorithm**

Charu Gandhi, Sayed Sayeed Ahmad , Abolfazl Mehbodniya , Julian L. Webber , S. Hemalatha, Haitham Elwahsh, and Basant Tiwari 

Research Article (14 pages), Article ID 4454226, Volume 2022 (2022)

## **An Integrated Approach for Cancer Survival Prediction Using Data Mining Techniques**

Ishleen Kaur , M. N. Doja, Tanvir Ahmad , Musheer Ahmad , Amir Hussain , Ahmed Nadeem , and Ahmed A. Abd El-Latif 

Research Article (14 pages), Article ID 6342226, Volume 2021 (2021)

## Research Article

# Employing Atrous Pyramid Convolutional Deep Learning Approach for Detection to Diagnose Breast Cancer Tumors

**Ehsan Sadeghi Pour, Mahdi Esmaeili , and Morteza Romoozi**

*Department of Electrical and Computer Engineering, Kashan Branch, Islamic Azad University, Kashan 8715998151, Iran*

Correspondence should be addressed to Mahdi Esmaeili; [m.esmaeili@iaukashan.ac.ir](mailto:m.esmaeili@iaukashan.ac.ir)

Received 20 May 2022; Revised 8 October 2022; Accepted 24 November 2022; Published 14 November 2023

Academic Editor: Mohammed A. A. Al qaness

Copyright © 2023 Ehsan Sadeghi Pour et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Breast cancer is among the most common diseases and one of the most common causes of death in the female population worldwide. Early identification of breast cancer improves survival. Therefore, radiologists will be able to make more accurate diagnoses if a computerized system is developed to detect breast cancer. Computer-aided design techniques have the potential to help medical professionals to determine the specific location of breast tumors and better manage this disease more rapidly and accurately. MIAS datasets were used in this study. The aim of this study is to evaluate a noise reduction for mammographic pictures and to identify salt and pepper, Gaussian, and Poisson so that precise mass detection operations can be estimated. As a result, it provides a method for noise reduction known as quantum wavelet transform (QWT) filtering and an image morphology operator for precise mass segmentation in mammographic images by utilizing an Atrous pyramid convolutional neural network as the deep learning model for classification of mammographic images. The hybrid methodology dubbed QWT-APCNN is compared to earlier methods in terms of peak signal-to-noise ratio (PSNR) and mean square error (MSE) in noise reduction and detection accuracy for mass area recognition. Compared to state-of-the-art approaches, the proposed method performed better at noise reduction and segmentation according to different evaluation criteria such as an accuracy rate of 98.57%, 92% sensitivity, 88% specificity, 90% DSS, and ROC and AUC rate of 88.77.

## 1. Introduction

Breast cancer occurs in the breast and has symptoms such as a lump in the breast, breast appearance changes, breast skin dimpling, nipple discharge other than breast milk, and/or flaky skin. Breast cancer is the second-most frequent cancer among women and causes a large number of deaths every year. It was reported that breast cancer is almost impossible to prevent since its causes remain unknown [1]. Therefore, early diagnosis is crucial in the treatment of breast cancer. Mammography is widely used by radiologists to diagnose and screen breast cancer. Today, mammography is the most commonly used technique for the early diagnosis of breast cancer and has reportedly lowered the mortality rate to 25%. However, it is difficult to interpret and describe mammographic images [2]. To obtain more accurate results, image preprocessing is required [1]. Preprocessing is primarily

carried out to enhance image quality and improve diagnosis by removing unimportant segments from the background and to precisely extract breast areas by revealing breast boundaries [2]. The current mammography is based on smart medical diagnosis systems with image processing using machine learning (ML). Image processing principles in smart medical systems are important for the diagnosis of breast cancer since mammographic pictures are intrinsically noisy, which may challenge the diagnosis. In reference [3], a number of optimal filters have been introduced in order to detect sounds. Although intelligent diagnosis systems can remove noise and detect diseases, the judgment of doctors is necessary. Therefore, it is important to introduce an intelligent diagnosis system to diagnose breast cancer.

In the proposed approach, a dataset called MIAS is used as the input dataset containing images and features of mammography for breast cancer diagnosis. This study is

mainly based on image processing and deep learning techniques. In other words, an image is first used as the system input. It is then preprocessed through the quantum wavelet transform algorithm for noise reduction. Morphological processing is then performed with expansion, erosion, and border operators as well as segmentation operations for feature detection. Afterwards, the image and its features are used as the convolutional deep learning network input, and the windowing order is performed in the network by layering. Feature extraction is then presented with classification. The Atrous pyramid CNN was employed in order to prevent classification problems. The results indicated that the proposed approach improved the cancer type diagnosis accuracy as opposed to the most of previous methods. In this study, a morphology-based quantum wavelet transform approach was employed to improve and reduce noise. In fact, this quantum wavelet transform is among the wavelet transforms that operate faster in detecting noisy areas. Due to its quantum mode, this wavelet transforms benefits from a higher processing speed to detect any noise on mammography images. There are certain advantages and disadvantages to each of the previous papers and studies. For instance, most of them did not use real-time processing but had high computational complexities and long runtimes. Basically, they had an uncertainty structure, and their final diagnosis accuracies were lower than the results reported by this study, in which all of the aforementioned metrics were improved. In each research step, the proposed approach was compared with previous methods, something which indicated the superiority of the research results. In summary, this study presents a method based on image morphology operators for the segmentation of mammographic pictures with the goal of detecting the precise mass area.

## 2. Literature Review

Since the intelligent diagnosis of breast cancer is a hot topic, numerous studies have been conducted using different methods in the literature. This section reviews the literature and the idea. This is classified into (1) breast tumor detection and classification, including the noise reduction of mammographic images, and (2) mammographic image diagnosis and classification.

*2.1. Noise Reduction-Based Studies.* The noise of mammographic images substantially affects image analysis and classification accuracy. Hence, it is important to reduce noise in mammographic images. The noise of a medical image is dependent on the imaging procedure. Mammographic images often have Gaussian, impulse (salt and pepper), and Poisson noises. Such noises should be minimized to avoid challenges in the next processing phase and breast tumor misdiagnosis.

*2.1.1. Salt and Pepper Noise.* Salt and pepper noise appears in the form of corrupted white and black pixels, which could be sparse or dense. It is also known as impulse noise and

often occurs in data transmission. Abrupt disruptions in the image signals are the main cause of salt and pepper noise. It has two scenarios of probability: zero or 255 (eight-bit images); it either makes a signal zero (destruction) or one (the noise replaces the signal) [4].

*2.1.2. Gaussian Noise.* Gaussian noise, sometimes known as white noise, typically arises from electric sensors to capture image signals. It is based on the Gaussian distribution that is randomly selected and applied to the image. The Gaussian noise measure of a Gaussian distribution is given by

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(z-\bar{z})^2/2\sigma^2}, \quad (1)$$

where  $z$  is the gray level,  $\bar{z}$  is the mean gray level, and  $\sigma$  is the standard deviation. Here,  $z$  and  $\sigma$  are the mean and variance of the Gaussian distribution, respectively [5].

*2.1.3. Poisson Noise.* Poisson noise, also known as quantum mottle in medical physics, occurs in images due to Poisson processes. It arises from the distinct properties of photons. It appears between the original pixels in an image in a dispersed form. Poisson noise is found between the high-frequency components of an image [6].

A study on noise reduction from mammographic images [7] found that the level of noise significantly affected image analysis and classification. It is, therefore, important to reduce noise in mammograms. Medical images have different amounts of noise. Quantum noise is the most prevalent type of noise in mammography imaging. The goal of this research was to identify and investigate various filters in windows, including mean, middle, and Wiener filters of various sizes, using the DDSM (Digital Database for Mammography Screening) dataset. The greater the noise rating is, the higher the peak signal-to-noise ratio (PSNR) is, implying that the restored image has a higher image quality. The PSNR value was used to analyze the image quality of the restored filters. According to the results, for the reduction of noise in mammographic images, the  $3 \times 3$  Wiener filter produced the best results.

In another study [8], to reduce noise in grating-based mammographic images using X-ray, nonlocal denoising based on noise analysis was used. Noise analysis-based nonlocal denoising methods use noise variance similarity and dispersion to obtain the optimal weighted average using pixel intensity. The noise variance was calculated more accurately using a two-stage NLM-NANLM method. The method showed superb performance.

A study presented a preprocessing technique for mammograms using an adaptive weighted frost filter [9]. Mammography is the best successful technology for the initial detection of breast cancer in patients since it can identify cancer two years before symptoms appear. The pre- and postprocessing stages of the mammographic image identification procedure are computationally intensive. In all imaging approaches, initial processing is critical, with the most critical component being the implementation of techniques capable of enhancing the image's quality so that it

can be used for further analysis and data extraction. This article discussed preprocessing, which is critical in mammographic picture analysis due to the low quality of mammography, which is taken at low doses since excessive radiation can threaten the patient's health. Numerous strategies have been developed to enhance image quality, smoothness, and noise restoration. The experimental results indicated that the suggested adaptive weight freeze filter is the optimal solution for noise reduction in mammographic pictures, outperforming other methods. The proposed technique was compared qualitatively and quantitatively to the other strategies available. The article's experiments demonstrated that the proposed strategy outperforms previous techniques.

In another study [10], the Bayes shrink (HMBS) method was introduced in order to reduce speckle noise in mammographic images. A combination of homogenous filters and downsized methods was used to reduce Bayes for denoising. Homogeneous filters were used to differentiate between homogenous areas and speckle noise, and seven criteria were employed to more accurately evaluate image quality.

In reference [11], radiologists require high-quality and perfect mammographic images for more accurate diagnosis. Using convolutional neural networks (CNNs) as a deep learning model, a method for reducing noise in images and improving diagnosis was proposed. Poisson noise was increased, and ensemble transmission was used to convert it into white Gaussian noise. Moreover, the authors in [12] describe the development of an intelligent breast cancer detection system. This unique strategy is based on the use of image processing techniques to extract the tumor area while taking into account its significant characteristics. Then, seven features representing the tumor's texture and shape are retrieved and fed into a back-propagation neural classifier. The researchers also proposed the use of an interval type-2 fuzzy set and HM approach to fuzzify a breast cancer dataset [13]. They used the Wisconsin Breast Cancer dataset from the UCI data source for the purpose of creating the fuzzy breast cancer dataset. To overcome the limitations of the classic fuzzy type 1 method, the IT2 fuzzy models captured several expert opinions that addressed sharp boundary problems as well as inter- and intra-uncertainty among domain experts. By utilizing this database, rules and models will be developed that are more accurate.

*2.2. Segmentation-Based Studies.* Dissecting malignant masses in mammograms is a difficult task when there are issues such as low contrast, ambiguous, hazy, or divided boundaries, and the presence of severe abnormalities [14]. These facts exacerbate the difficulty of developing computer-aided diagnostics (CAD) tools to assist radiologists. The purpose of this article [14] was to offer a new mass separation algorithm for mammography based on robust multifunctional characteristics and automatic and maximal estimation (MAP). Four steps were proposed as part of the segmentation approach: a dynamic contrast enhancement strategy that applies to a specified region of interest (ROI),

a technique for correcting background infiltration using matching templates, and mass candidate point recognition using posterior probabilities based on various scales.

The high degree of integration and the precise specification of the mass area are achieved through a MAP system in image segmentation. Segmentation was performed using 480 ROIs created in collaboration with two radiologists and ground truth. Three statistical criteria were utilized to assess its effectiveness in comparison with advanced segmentation techniques. The experimental results demonstrated that the created approaches are capable of comparing to other algorithms for ill-defined or thicker masses. By incorporating it into a CAD system, radiologists may benefit from this strategy.

The authors of [15] present a method for classifying and diagnosing breast cancer in mammographic pictures using a mix of wavelet analysis and a genetic algorithm. As presented in this paper, concerns have been raised about the reliability and sensitivity of detecting abnormalities in both lateral oblique and cranial-ear (CC) mammographic views. This study discussed a group of computational algorithms for identifying and segmenting mammograms with or without masses in the CC and MLO images. To begin, an algorithm for removing artifacts was run utilizing a wavelet transform and Wiener filter-based approach for gray-level enhancement. Additionally, a method has been presented for identifying and dividing masses randomly selected from the digital mammography screening dataset using genetic algorithms, multiple thresholds, and wavelet transforms genetic algorithms. An area overlap metric (AOM) was used to test the computer approach developed. Experiments demonstrated that the proposed method could be used to segment mammography masses in CC and MLO images. Additionally, this strategy overcame the examination of the CC and MLO representations.

Additionally, another study [16] proposed a semisupervised fuzzy GrowCut adaptive method of segmenting mammographic pictures based on the region of interest. In the study, the automaton evolution rule was modified to include a Gaussian fuzzy membership function in order to model undefined borders in a semisupervised version of the GrowCut algorithm. As part of this method, the manual selection of suspected lesion locations was replaced with an automated selection process that utilized a differential evolution algorithm only to select interior points. 57 lesion photos from the mini-MIAS database were used to assess this approach. The results were compared to those obtained using LBI, wavelet analysis, BMCS, BEMD, MCW semisurveillance, and the topographic technique. The results indicated that the method produced superior results for hybridized, thicker, and poorly acquired lesions due to the relation between the images of the grand tract and the segmentation results. In reference [17], using two fully convolutional neural networks (CNNs) based on SegNet and U-Net, two deep learning strategies were proposed for the automated segmentation of breast tumors in dynamic contrast-enhanced magnetic resonance imaging (DCE-MRI). The advantage and superiority of the proposed method in this study are its high accuracy in the division method for better and more accurate identification of the masses.

In another study [18], earlier works developed a deep learning system to detect and diagnose breast cancer in mammographic images based on the end-to-end strategy. A transferable texture (TT)-CNN-based classification method was employed for cancer classification. The benign and malignant areas would be detected using the TT-CNN architecture once the mammographic images had been processed. Then, EL investigated the tissue features and extracted data from the image. For example, in reference [19], the U-net architecture was employed to segment fibroglandular tissue (FGT) and breast images. The model was demonstrated to substantially outperform other algorithms. A CNN was employed to segment mammographic images and find deep masses. In fact, a multipurpose segmentation was provided for different image areas. They demonstrated that an individual CNN architecture could be exploited to train other CNNs to obtain more accurate information from images using different methodologies [20]. For the segmentation of prostate and mammographic images, convolutional neural networks and deep learning have also been implemented. In this research [21], using the U-net model, breast lesions were segmented into two stages: U-net and quantity. The model was found to outperform other techniques and could be utilized for ultrasonic breast cancer detection and diagnosis. In another study [22], local adaptive thresholding and an advanced morphologic method were used for nuclear Allred cancer segmentation and classification in breast tissue images. They performed unsupervised classification of cancer nuclei. The model was calculated to have an accuracy of 98% in tumor-level measurement.

In reference [23], mammographic images were segmented to detect and classify cancerous tumor types (i.e., benign and malignant) from an optimal region growth perspective. The images would be noise reduction using a Gaussian filter prior to primary image processing. Drawing on the gray-level run length matrix (GLRLM) and gray-level co-occurrence matrix (GLCM) techniques on segmented images, tissue features were extracted and fed to a feed-forward neural network (FFNN). The tumors were classified into benign and malignant through a backpropagation (BP) algorithm. The model showed an accuracy of 97.8% and outperformed other models.

In [24], to detect and classify benign and malignant cancerous tumors, two automatic techniques were introduced: (1) the detection and classification of growing tumors, in which the threshold was obtained through a trained neural network, and (2) tumor detection and classification using a cellular neural network (CNN). The techniques were implemented on the mammographic image analysis society (MIAS) dataset, with the accuracy, sensitivity, and specificity being 95.94%, 96.87%, and 96.47%, respectively. A three-stage automatic system was proposed for the detection and classification of tumors using microarray images. The system was reported to have an accuracy of 95.45% [25]. An automatic backpropagation neural network (BPNN) model was introduced for the classification and detection of breast cancer tumors. It was reported to detect cancerous tumors with an accuracy of 70.4% [26]. The naïve Bayesian algorithm was adopted to

detect and classify cancerous tumors on mammographic images. The accuracy, sensitivity, and specificity of the algorithm were reported to be 98.54%, 99.11%, and 98.25%, respectively [27]. In another study [28], a personal mammographic screening method was developed to diagnose cancer breast on mammographic images. It implemented screening decision-making based on the age of a patient. In reference [29], a hybrid predictor of breast cancer recurrence was employed. The model was calculated to have an accuracy of 85%. In reference [30], a hybrid of the firefly algorithm and artificial intelligence (AI) was employed to detect breast cancer. In another study [31], AI and image-processing techniques were employed to detect breast cancer. Furthermore, a new breast cancer detection methodology was introduced using ML algorithms. In reference [32], an automatic system was proposed for breast cancer classification. They used deep learning for the classification and detection of cancer on ultrasound images. The technique consisted of five phases: (1) data enhancement, (2) a pretrained model, (3) training the modified model through transfer learning (TL), (4) selecting the best features, and (5) the classification of the selected features using ML.

In another study [33], bat-inspired algorithms (BA) can be utilized for cancer classification using microarray datasets for gene selection. Two stages are employed in gene selection, namely, the filter stage that utilizes the minimum redundancy maximum relevance (MRMR) method and the wrapper stage that utilizes BAs and SVMs. In this paper, the authors in [34] proposed a methodology to detect breast cancer and classify malignant and benign tumors. To extract features from mammogram images, ML and hybrid thresholding were employed. The model was evaluated on four mammogram image datasets, including MIAS, DDSM, INbreast, and BCDR. The model was found to show maximum performance on the MIAS dataset.

In reference [35], a new feature learning approach was proposed to detect and classify breast cancer using an artificial neural network (ANN) with optimized hidden layers. The classification sensitivity, accuracy, and specificity were reported to be 0.9815, 0.9948, and 0.9882, respectively. In this review [36], earlier works reviewed the literature on kidney cancer detection and the classification of malignant and benign tumors using ML and deep learning algorithms. In reference [37], the literature on breast cancer detection and classification based on ML algorithms was reviewed. The detection of breast cancer on mammographic images is carried out in three stages: (1) image preprocessing, (2) feature extraction, and (3) classification and evaluation. A total of 93 works were reviewed, reporting that deep learning techniques account for the majority of the effective methods that are used for cancer detection.

### 3. Proposed Method

The present study primarily aimed to implement the early detection of breast cancer on mammographic images and tumor classification into benign, malignant, and suspicious using a hybrid of image processing techniques and deep learning. Figure 1 demonstrates the proposed method diagram in which the operations of each step are presented briefly.

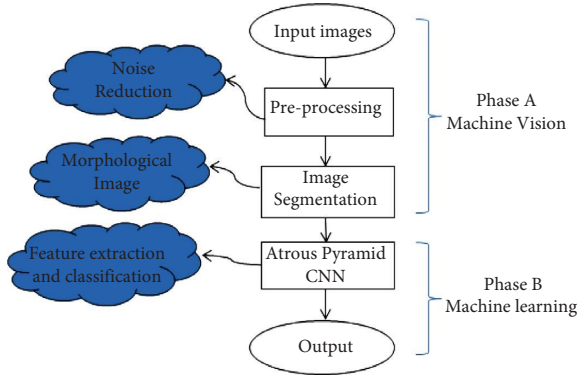


FIGURE 1: Proposed method diagram.

The proposed approach consists of three major steps, the first of which includes preprocessing to improve and reduce noise on mammography images through the quantum wavelet transform. In the second step, morphological processing is used for image segmentation. In fact, these two steps are considered the phase of image processing and machine vision. The third step operates with a deep learning structure based on an Atrous pyramid convolutional neural network (APCNN) that actually selects and extracts features in addition to performing classification operations to diagnose benign, malignant, and suspicious cases of cancer. It can also pinpoint the accurate locations of cancerous tumors on an image. This step belongs to the machine learning phase.

**3.1. Preprocessing Phase.** The input images should be normalized. In the preprocessing phase, mammographic images were normalized as input data (often noisy) and were improved to enhance system efficiency. The image is changed to a predefined size and reasonably filtered using quantum wavelet transform. Then, the input data are normalized. A two-dimensional array of pixels in the range  $[0, 225]$  is used to display individual images in a hybrid of local thresholding and active contours. The local thresholding process initializes images in two stages. It is assumed that the noisy input image will be the initial image for denoising. This is carried out by local search operators to improve the initial pictures using quantum wavelet transform. Therefore, following the first phase, a deconstructed image will exist. The second stage involves thresholding the detail coefficients and randomly selecting one of these decomposed regions for reconstruction. The following definitions apply to the reconstruction section:

- (i) Gauss fading: filter image using a Gaussian filter
- (ii) Means filter: filter image using a mean filter
- (iii) Change in intensity: a similar criterion is chosen at random between  $[0.7, 1.3]$  to multiply all picture pixels
- (iv) Adaptation of light intensities: an inverse quantum wavelet transform filtering technique based on quantum and inverse processing is used to construct the quantum inverse structure

The following steps will then be taken:

- (i) One-point row: randomly selected pixels in a row
- (ii) One-point column: randomly selected pixels in a column
- (iii) Point-to-point pixel: as each pixel disappears, it is replaced by a random pixel
- (iv) Classifying all the points as rows and columns in the pictures and diagonally to decrease the noise in quantum wavelet transform

In the quantum wavelet transform filtering algorithm, a new picture may be passed through the local search operator when the selection value is less than the range  $[0, 1]$  lower than the local search rate. Each pixel in the image is sorted by its pixel value after the decomposition process has been completed, and the best coefficients are used as quantum values for the operation at hand. There are several ways to decompose a signal in mammographic pictures into several displaced or scaled displays of previously extracted characteristics. In order to break down an image into its constituent components, local thresholding and active contours can be applied. After applying the quantum wavelet transform, local thresholding, and active contours, the image is segmented. Some details can be eliminated by applying quantum wavelet transform-based local thresholding and active contour coefficients. Local thresholding and QWTF based on active contour provide the significant advantage of distinguishing small features in an image. It is possible to isolate very small details in an image using active contours, while larger details can be detected using local thresholding. The combination of small and large details and reading all rows and columns linearly and diagonally meet the quantum wavelet transform structure so that mammographic image noise can be minimized. Two characteristics are present in a local thresholding-active contour function with quantum wavelet transform. First, it is a vibrational function or has a wave-like form, such as follows:

$$\int_{-\infty}^0 \Psi(t)|^2 dt < \infty. \quad (2)$$

The maximum energy in  $\Psi(t)$  occurs in a limited period, which is written as follows:

$$\int_{-\infty}^0 \Psi(t)dt = 0. \quad (3)$$

Reducing the noise method is written as follows:

$$\text{Method}(I) = \left( \sum_{\Omega} \sqrt{1 + \alpha^2 |\nabla I|^2} \right) + \frac{\gamma}{2} (I - I_0)^2. \quad (4)$$

In this function, the image edges are taken into account, and important characteristics of the image are preserved. The term  $(I - I_0)^2$  ensures a specific degree of validity between the picture under evaluation and the original picture, in which  $I$  and  $I_0$  represent the picture under study and the original picture, respectively. Furthermore,  $\nabla I$  is the total diversity tuning period,  $\alpha$  and  $\gamma$  are balancing parameters, and  $\Omega$  is the total of pixels in the picture. The minimization

of equation (3) reduces the total picture diversity while preserving validity. Overall, input data are normalized in the preprocessing phase and improved, if needed, to enhance the detection performance of the system.

It is important to note that by adjusting the sum of the variations  $\nabla I$ , a mammography picture may have some noise such as salt and pepper, Gaussian, or blur effects. Therefore, this variation was used to determine the types of this noise variation and to calculate its sum. In this article, QWTF is proposed as an innovative noise reduction method for mammography. Earlier works adopted the matched filter technique to introduce a strategy to detect macroscopic dark material objects in images [38], and also, a quantum image filter in the frequency domain was introduced based on the Fourier transform [39]. It should be noted that the threshold value was determined by trial and error. Figure 2 illustrates how to identify noisy pixels.

For identifying noisy pixels in figures, each pixel has four brightness values ranging from white to black, and these values are  $\text{pos} = |01\rangle$  and  $\text{color} = |10\rangle$  for dark gray,  $\text{pos} = |00\rangle$  and  $\text{color} = |01\rangle$  for gray color,  $\text{Pos} = |11\rangle$  and  $\text{color} = |11\rangle$  for white color, and  $\text{Pos} = |11\rangle$  and  $\text{color} = |11\rangle$  for black color.

**3.2. Image Segmentation Phase.** The segmentation of images is one of the most important and complex parts of image processing and computer vision. Today, segmentation is a standard image processing and manipulation process in many software packs and systems. In this process, similar pixels are segmented into the same class. In other words, images are partitioned into sections or objects. To effectively identify the image space, it is required to identify the foreground and background. To this end, internal edge detection is used, and different segments of an image can be separated in terms of color and light based on edges. The input of the segmentation phase consists of images that have been denoised and improved in the preprocessing phase. The operation is carried out based on the morphology in the segmentation phase. This algorithm is used for two reasons. First, an image is assumed to be a search space, and segmentation can be used to improve the search space. This effectively reduces dimensionality, extracts features, and implements classification to enhance performance. Second, it boosts the speed and convergence of image processing and avoids local optimal. It is worth mentioning that edge detection based on the Sobel operator is also utilized. In this respect, MATLAB has preprocessing instructions.

Mathematical morphology helps extract image components, which is very useful for describing segment features and shapes, such as frameworks, convex shells, and boundary areas. The mathematical morphology language is set theory, and morphology is a powerful, unified technique to cope with image processing problems. Here, sets represent objects in an image. Erosion and dilation are the two essential operations in morphological image processing. A segmentation phase is performed to segment mammographic images using morphology based on erosion and dilation operations and boundary extraction. Let  $M$  and  $N$  be sets in  $q$ . The erosion of  $M$  and  $N$  is written as follows:

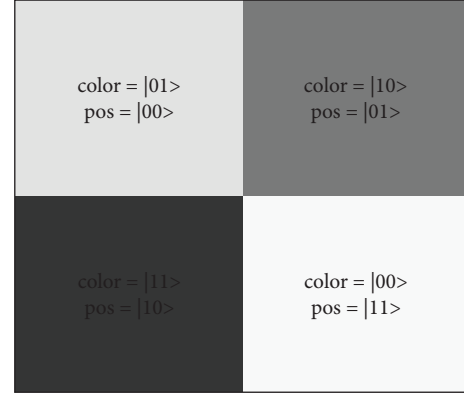


FIGURE 2: Identifying noisy pixels [38].

$$M \ominus N = \{Q|(B)_q \subseteq M\}. \quad (5)$$

The erosion of  $M$  and  $N$  is a set of all points of  $q$  such that  $N$  transferred by  $q$  is located in  $M$ .  $N$  is assumed to be a structuring element. Since  $N$  should be in  $M$ , set  $N$  and the background share no objects. Erosion can also be formulated as follows:

$$M \ominus N = \{Q|(N)_q \cap M^c = \emptyset\}, \quad (6)$$

where  $M^c$  is the complement of  $M$  and  $\emptyset$  is the empty set. Let  $M$  and  $N$  be set in  $Q^2$ . The dilation of  $M$  and  $N$  is written as follows:

$$M \oplus N = \{Q|(\hat{N})_q \cap M \neq \emptyset\}, \quad (7)$$

where dilation is implemented by reflecting  $N$  around its origin and transferring the reflection by  $q$ . Then, the dilation of  $M$  by  $N$  is the set of all movements in  $q$  such that  $\hat{M}$  and  $\hat{N}$  have at least one common element. Therefore, dilation is formulated as follows:

$$M \oplus N = \{Q|[(\hat{N})_q \cap M] \subseteq M\}. \quad (8)$$

Based on equation (8),  $N$  is a structuring element and  $M$  is a set of image objects to be dilated. The boundary of set  $M$ , shown as  $\beta(M)$ , can be found by eroding  $M$  by  $N$  and subtracting  $M$  from its erosion as follows:

$$\beta(M) = M - (M \ominus N). \quad (9)$$

Based on equation (9),  $N$  is a suitable structuring element.

For calculating the fitness function of  $f$  in the proposed image morphology operators in this article, the dataset considers as  $R_{N \times D}$  which  $N$  is the sample from per image and  $D$  is the sample's distance (features) which will have  $K$  segmented parts and  $f$  calculated as follows [40]:

$$f = \text{minimize} \sum_{i=1}^N \sum_{m=1}^K [\delta(r_{i,d}, c_{m,D})]^2. \quad (10)$$

In this equation,  $\delta$  is distance (features) metric as Euclidian between any segmented parts which is defined based on two features: brightness and edges.

**3.3. Atrous Pyramid CNN.** This section presents APCNN based on convolutional neural networks as a new method of deep learning that can simultaneously calculate features and classify data. However, for the purpose of this research, it is intended that it would be able to determine breast cancer and then to identify the exact area of the masses followed by classifying them into malignant, benign, and suspicious classes. This section will be called APCNN which optimizes CNN with Moore–Penrose matrix and also CNN with this matrix. There are two general disadvantages associated with most neural network structures that can be addressed by adjusting the weights in the training phase using the descending gradient, as well as the volume of training data, in contrast to the classic CNN method. A further weakness in neural networks is a slowdown in the training process. This weakness can be resolved quickly during the training and testing phases, which is abundant after considerable data have been gathered. Neural networks also do not have the capability to train and test the same data if a similar dataset is imported or new data is entered into the same dataset which is another weakness of neural networks that is named generalization. Thus, there are many different types of neural networks that cannot be generalized. For this study, we will focus initially on CNNs.

It is interesting to note that in this study, the CNN will be optimized as an APCNN so that it can be run rapidly with generalizability and that is a result of the difficulties associated with neural network structures. As a result of its high learning speed and ability to adjust a parameter during the training phase as opposed to adjusting a number of parameters during the training phase in neural networks, this algorithm is often used. One of the major disadvantages of CNN is its inability to perform normal extraction, feature extraction, and classification operations. However, it will be performed by optimizing CNN and building APCNN structures. A CNN is a neural network that involves the input layer attached to a series of weights for the hidden layer, which are initially assigned a random value and are not reset during the training process, which is time-consuming. Unlike conventional neural networks, CNN uses normal neurons in the hidden layer; therefore, it does not require centroids and sigmas. Finally, there is only one parameter that needs to be adjusted in the CNN: synaptic weights between hidden and output layers. A typical CNN is a feed-forward structure that calculates synaptic weights in real time using an inverse pseudostructure, resulting in faster data training and testing. The overall architecture can be seen in Figure 3.

The most important reasons for using CNN in this study instead of other smart methods in the classification and feature extraction phase are shown in Table 1.

CNN, in general, can be viewed as the exact opposite of deep learning methods and other classification methods such as naive Bayesian and SVM methods. Due to the algorithm's tremendous flexibility, it can use nonlinear activation functions such as sinusoidal, sigmoid, or nonderivative activation functions in addition to linear activation functions to neurons or activate cells in the hidden layer. By default, CNN has an equation in the general mode such as follows:

$$z(p) = \sum_{j=1}^m \beta_j \beta_j g \left( \sum_{i=1}^n w_{i,j} x_i + b_j \right). \quad (11)$$

According to this equation,  $\beta_i$  represents the weights between the input layer and the hidden layer, and  $\beta_j$  represents the weights between the output layer and the input layer.  $b_j$  is the threshold value of neurons in the hidden layer, or bias.  $g(\dots)$  is the transition or actuator function.  $w_{i,j}$  is the input layer weights, and  $b_j$  is the bias that are assigned at random. At the start of the number of input layer, neurons,  $n$ , and hidden layer neurons,  $m$ , the activation function  $g(\dots)$  is assigned. According to this knowledge, if the known parameters for overall balance are merged and calibrated, the output layer will resemble as follows:

$$H(w_{i,j}, b_j, x_i) = \begin{bmatrix} g(w_{1,1}x_1 + b_1) & \dots & g(w_{1,m}x_m + b_m) \\ g(w_{n,1}x_n + b_1) & \dots & g(w_{n,m}x_m + b_m) \end{bmatrix}, \text{ and} \\ z = H\beta. \quad (12)$$

The main goal in all models of training-oriented algorithms is to minimize the error whenever possible.  $z_p$  is a function that outputs errors obtained by the actual output  $z_{\text{main}}$  in CNN, which can be represented by two training sections, namely,  $\sum_k^s (z_{\text{main}} - z_p)$  and testing sections, namely,  $\|\sum_k^s (z_{\text{main}} - z_p)^2\|$ . For both functions, the output  $z_p$  obtained by the actual output  $z_{\text{main}}$  must be equal to  $z_p$ . An unknown parameter is specified when this equation is executed, and the results are satisfied. The matrix  $G$  can be a matrix that is very unlikely. As a result, there may be a discrepancy between the whole number of attributes in the training and those in the test set. Therefore, inverting  $[G]$  and locating weights are important issues. CNN overcomes this challenge by using a matrix referred to as Moore–Penrose, which can be used to develop approximate inverse matrix computations that are capable of performing dimensionality selection and feature extraction operations with classification with increased accuracy and speed in comparison to other methods. Using the Moore–Penrose matrix,  $\alpha^*$  is the output matrix and  $G^*$  is the generalized inverse Penrose matrix of  $G$ . Thus, due to the optimization of the CNN, the problem of output weights in the CNN was solved as  $A^* = G^*$  which became the APCNN or Moore–Penrose matrix extreme learning machine. Generally, APCNN becomes a chain of repeating modules over time in the training phase. APCNN will be able to work like a conveyor that is to add or subtract information from neurons. APCNN does not require weight updating during training, unlike deep learning structures or other classification models such as naive Bayesian models and support vector machines. Unlike deep learning structures and other classification models, such as support vector machines or naive Bayesian, no weight update operations are performed during training. APCNN can specify attributes at the intersection. By minimizing APCNN energy performance, a suitable model is taught that can be modeled as follows:



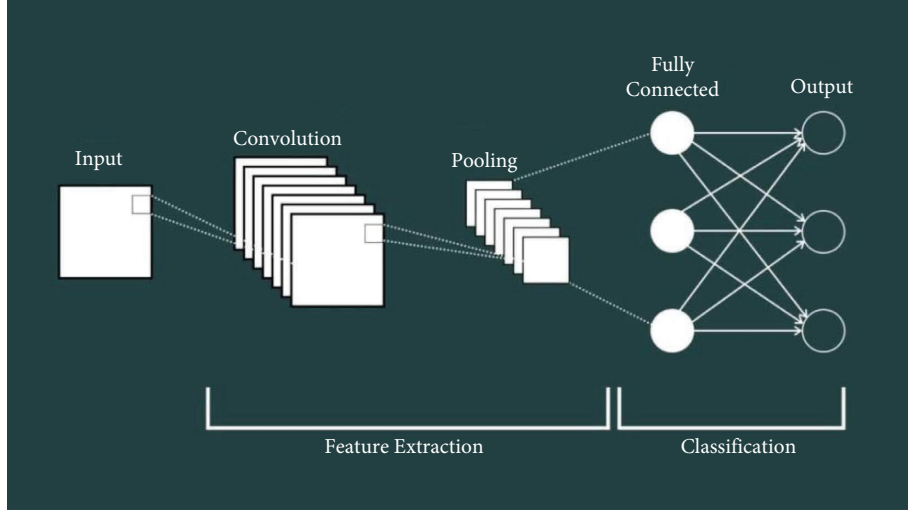


FIGURE 3: CNN's main architecture.

TABLE 1: Comparison of APCNN with other conventional intelligent methods.

Naïve Bayesian	Support vector machine (SVM)	Extreme learning machine (ELM)	Recursive neural network (RNN)	Convolutional neural network (CNN)	APCNN
Training process is slow	Training process is slow	Training process is slow	Training process is slow	Faster training process	Faster training process
Binary classification ability	Binary classification ability	Multiclass classification ability	Multiclass classification ability	Multiclass classification ability	Multiclass classification ability and multiobjective and real time
Quadratic programming	Quadratic programming	Nonlinear multiobjective programming	Multiobjective quadratic programming	Nonlinear multiclass	Nonlinear multiclass and multiobjective programming

$$E(Y) = \sum_i^N \Psi_v(y_i) + \sum_{v_i, i \neq j}^N \Psi_q(y_i, y_j). \quad (13)$$

In this case,  $v, q \in \{1, 2, \dots, C_n\}$  are the intersection labels, and  $i, j \in \{1, 2, \dots, N\}$  are specific pixels of the original image or  $I$ .  $\Psi_q(y_i) = -\log P(y_i | I)$  is a negative logarithmic probability in which  $P(y_i | I)$  is a probability calculated by the APCNN algorithm for each pixel  $I$ . As part of the evaluation of two APCNN matrices in a fully connected layer, it is necessary to examine the relationship between each pair of pixels outlined in the following equation:

$$\Psi_q(y_i, y_j) = \eta(y_i, y_j) \sum_{n=1}^N w^{(n)} k^{(n)}(f_i, f_j). \quad (14)$$

In this equation,  $N = 2$  is the number of Gaussian core and  $w^{(n)}$  indicates a weight for the  $m$ th Gaussian core.  $\eta(y_i, y_j) = [y_i \neq y_j]$  is the consistent function tag.  $k^{(1)}$  demonstrates the appearance of the core appearance, which attempts to assign the same class tags to adjoining and similar intensity pixels adjacent to each other.  $k^{(2)}$  demonstrates the core smoothness, which is connected with the objective of removing superfluous parts. Overfitting and data redundancy may occur within the max-pooling layer in matrix convolution deep learning. Generally, these problems are common in neural networks, especially in matrix

convolution deep learning. Hence, matrices were used in this study to prevent these problems and accelerate training and testing for the detection and extraction of features. These two steps are denoted by equations (15) and (16), respectively.

$$k^{(1)}(f_i, f_j) = \exp \left( -\frac{|s_i - s_j|}{2\theta_\alpha^2} - \frac{|e_i - e_j|}{2\theta_\beta^2} \right), \quad (15)$$

$$k^{(2)}(f_i, f_j) = \exp \left( -\frac{|s_i - s_j|}{2\theta_\gamma^2} \right). \quad (16)$$

$e_i$  and  $e_j$  are the light intensities of the pixels  $i, j$ ,  $s_i$ , and  $s_j$  of the corresponding spatial coordinates.  $f_i$  and  $f_j$  display the characteristics of each pixel pair, i.e., the brightness intensity and spatial information.  $\theta_\alpha$ ,  $\theta_\beta$ , and  $\theta_\gamma$  show the parameters of the Gaussian cores, respectively. However, some points may not be cut in this way; therefore, an optimization of this algorithm will be done in layers. Generally, the layers of the APCNN method are employed by using the input layer with the number of neurons. As part of the training and testing layer, convoluted layers, pooling layers, and fully connected layers have been implemented along with Moore–Penrose. Next, a soft-max layer and an output layer are then embedded in order to display the results. Matrix-based windowing is used for the training layers as

measured by  $9 \times 9$  in the convolve layer,  $7 \times 7$  in the random pooling layer, and  $5 \times 5$  in the maximum pooling layer. The fully connected layer's structure is CRF, and its window structure is  $9 \times 9$ . The soft-max layer is also  $7 \times 7$ . As part of the initial APCNN training and segmentation process, convolve and pooling layers are sequentially inserted into the training layer, which consists of a convolve layer, a random pooling layer, another convolve layer, and finally a maximum pooling layer. There is a completely connected APCNN layer at the conclusion of this training layer. Then, outside the training layer, there is a soft-max layer, which is used to optimize specification operations and motion object tracking following feature extraction using the probabilistic particle filtering technique. It is important to keep in mind that the amount of neurons in each segment is critical. For every convolve and pooling layers, there are seven Atrouses ( $r$ ). In order to enhance the segmentation and feature extraction activities during the training of the deep neural network of the soft-max layer, the APCNN method is applied. The state of a dynamic system can be approximated using Bayesian filters based on a sequence of sensory observations with noise. To begin, the most widely known Bayesian rule is that a probability for an APCNN technique is eliminated (thus, the name Atrous pyramid), whose model is the following equation:

$$p(C | D) = \frac{p(D | C) \times p(C)}{p(D)}. \quad (17)$$

In these relationships,  $p(R(m)|S(m))$  is a new estimation,  $W(n)$  is scaling,  $p(S(m) | R(m))$  are probably observations of a motion object, and  $p(R(m) | S(m-1))$  is the probability before observing the tumor masses based on sentinel lymph nodule, metastasis, and assessment of mitotic density. Also,  $p(R(m-1) | S(m-1))$  is the preceding estimation, and  $p(R(m) | R(m-1))$  is system dynamics in the detection of tumor masses. Now, assuming that the  $S(m)$  are independent of one another, the system is described as a probabilistic APCNN process. By and large, the proposed Bayesian models are quite intricate, and it is difficult to study Gaussian distributions, at least in terms of linear models. While relationships can be simplified to achieve the required level of deep learning, generally, in order to solve equations, probabilistic APCNN techniques are used to consider all possible variations.

Probabilistic APCNN has as its primary objective to determine the conditional density probability function for the mode vector and the measurement vector, and to apply

If Bayesian procedures are used to update the H assumption under the premise of  $E$  and  $I$ , there is the following equation:

$$p(H | K, L) = \frac{p(K | H, L) \times p(H | K, L)}{p(K | L)}. \quad (18)$$

In this case,  $p(K|H, L)$  is the likelihood of the subsequent occurrence of  $H$  assumption based on the assumption of observing  $E$  in test conditions  $L$ .  $p(H|K, L)$  indicates the likelihood that the  $H$  assumption will take place prior to the  $L$  test conditions and the  $E$  perspective. Rate of similarity  $p(H|K, L)$  indicates the likelihood that the  $K$  assumption will occur when the  $H$  hypothesis meets the  $L$  test conditions and, lastly, the  $p(K|L)$  criterion for homogenization. When all measurements and values are taken into account, it is assumed that  $S(m)$  up to and including  $m$  and the value of  $R(m)$  of a dynamic system at  $m$ th can be predicted. Alternatively, a probabilistic probability can be calculated using a Bayesian formula:

$$p(R(m) | S(m)), \quad (19)$$

so that  $S(m) = \{s(1), s(2), \dots, s(m)\}$  is the set of all observations, and similarly, the state set of values  $R(m)$  is defined as  $R(m)$ , and  $R(0)$  contains historical information about the system's status (before any observation). Bayesian rules thus become a type of the following equations:

$$p(R(m) | S(m)) = \frac{p(S(m) | R(m), S(m-1)) \times p(R(m) | S(m-1))}{p(T(n) | S(m-1))}, \quad (20)$$

$$p(R(m) | S(m)) = W(m) \times p(S(m)|R(m)) \times p(R(m)|S(m-1)), \quad (21)$$

$$p(R(m) | S(m-1)) = \int p(R(m)|R(m-1)) \times p(R(m-1)|S(m-1)) dr(m-1). \quad (22)$$

Bayesian theory without utilizing any linearization and just modeling the entire system dynamically. This is one of the Monte Carlo statistical approaches, whereby the distribution function corresponds to the conditional probability of the weighted sum of a number of discrete functions. There are several types of Bayesian filters, which are commonly referred to as Bayesian bootstrap filters. Bayesian filters enable the estimation of a mode vector element's function based on the minimum error variance. Apart from Bayesian concerns and theory, as a result of equation (23), the method particles are defined as probabilistic for use in the soft-max layer of the APCNN algorithm; it is a function of the normal distribution function in two-dimensional and three-dimensional spaces.

$$p(\bar{x}) = \frac{1}{2\pi\sigma_1\sigma_2} e^{-\left[\frac{(x_1-\mu_1)^2}{2\sigma_1^2} + \frac{(x_2-\mu_2)^2}{2\sigma_2^2}\right]}. \quad (23)$$

APCNN can also identify and classify data into three categories: benign, malignant, and suspicious cancers.

#### 4. Simulation and Results

A MATLAB platform was used for simulation and analysis. A statistical analysis of the MIAS dataset has been used in this study. The characteristics of the data used in the MIAS dataset are clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, and mitoses. Based on the statistical data of this section, we will be able to accurately diagnose breast cancer, nonbreast cancer, and suspicious cases in this dataset. We may download this dataset at <https://peipa.essex.ac.uk/info/mias.html> link, which contains seven columns, as shown in Table 2:

The simulation is created step by step. As shown in Figure 4, when the simulation begins, the input image is executed and displayed.

As part of the preprocessing process, the first step is to reduce the picture size and make it identical with the original noise reduction by using a simple median filter to reduce noise. To reduce noise and improve the picture, the proposed quantum wavelet transform filtering method is then used, as shown in Figure 5.

According to statistical analysis, the proposed noise reduction approach has high capabilities in comparison to previous methods; Table 3 illustrates the evaluation criteria by case.

By pressing the segmentation with the image morphology operator button, the social spider algorithm performs the segmentation operation at a speed of 0.5 seconds, as shown in Figure 6.

It is necessary to define operators of the social spider algorithm segmentation algorithm for the initial population of spiders with 100 spiders, the blade vibration rate of 2 as standard, and the rate of prey attack as 0.02 as standard and to take into account the initial presentation of the algorithm as well. Segmentation is performed at 100 iterations, using both color and edge properties. On the basis of statistical analysis, the proposed algorithm has a high capability when compared with previous approaches to image segmentation. Table 4 shows a comparison of this approach to other methods in terms of evaluation criteria.

Subsequently, the morphology-based quantum wavelet transform algorithm was employed with the boundary operator for noise reduction in the segmentation stage. The noise was reduced as much as possible for the accurate zone detection and final classification, and Figure 7 depicts the output.

The deep convolutional neural network (CNN) is then used for two purposes: feature extraction and final classification. Therefore, the pyramid deep CNN is employed for feature extraction including dimensionality reduction and feature selection. Moreover, the Atrous deep CNN is utilized to classify and indicate masses accurately within a spectrum in the image. In fact, the pyramid CNN should be adopted for dimensionality reduction, feature selection, and feature

extraction based on the training and test models, in which 70% and 30% of data are used for training and test methods, respectively. There is a general output shown in Figure 8 that indicates only the breast. These operations are performed with the features introduced in Table 2 such as the column thickness, cell size uniformity, cell shape uniformity, marginal adhesion, single epithelial cell dimensions, naked cores, long chromatin, normal cores, mitosis, brightness, and edges. Furthermore, these features are used for the main research purpose that is to diagnose the metastasis of sentinel lymph nodes and assess mitotic density.

The classification operations are then performed by defining three classes (i.e., benign, malignant, and suspicious) and displaying the areas of cancerous tumors in mammography images, and Figure 9 indicates the output.

The operations in an input image have been displayed. However, all outputs should be implemented on a complete MIAS dataset. For this purpose, it is necessary to classify the analytical and statistical data of MIAS, which will be performed through the Atrous pyramid CNN. This method is adopted due to its simplicity among neural networks with a high convergence rate in training. However, it has some defects that can be covered with moving functions in addition to using a training core and the Atrous approach. Moreover, 70% and 30% of statistical data and images of MIAS were used for training and test methods, respectively. The Atrous pyramid CNN has nine major inputs with 10 hidden layers in the first layer and 2 hidden layers in the second layer. It also has two outputs called the detection of a tumor or mass in the breast or its absence. However, the third case known as the suspicious state was considered separately. If the output indicates neither the presence nor the absence of a tumor or a mass in the breast, it will be considered suspicious. Figures 10–13 demonstrate the efficiency, training modes, confusion matrix, and ROC of the Atrous pyramid CNN, respectively, and for breast cancer diagnosis based on MIAS images. Moreover, the ROC was used as the validation method along with K-fold and AUC.

Figure 14 depicts another diagram showing the accurate results of classification. This can be used to accurately diagnose breast cancer based on mammography images.

The 5 K-fold validation method was employed to draw outputs in Figure 14. It is evident that our method provides good results in the classification phase. 98.57% accuracy was obtained in this method. Table 5 reports the evaluation criteria for the proposed Atrous pyramid CNN. On the other hand, Table 6 shows the results of comparing this method with previous methods. The entire proposed approach should be represented as a ROC diagram from the beginning, i.e., preprocessing, segmentation, and then feature extraction and classification operations, and the output is in the form of Figure 15.

The final output, which completely extracts and displays the lesion or mass, is shown in Figure 16.

TABLE 2: The information available in the MIAS dataset.

#1	#2	#3	#4	#5	#6	#7
		Class of abnormality present: CALC, calcification CIRC				
MIAS database reference number	Character of background tissue: F, fatty G, fatty-glandular D, dense-glandular	well-defined/circumscribed masses SPIC, spiculated masses MISC, other, ill-defined masses ARCH, architectural distortion ASYM, asymmetry NORM, normal	Severity of abnormality; B, <i>benign</i> M, malignant	$x, y$ image coordinates of center of abnormality	$x, y$ image coordinates of center of abnormality	Approximate radius (in pixels) of a circle enclosing the abnormality

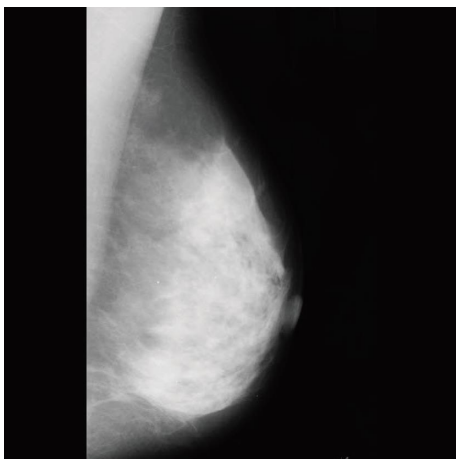


FIGURE 4: Input image.

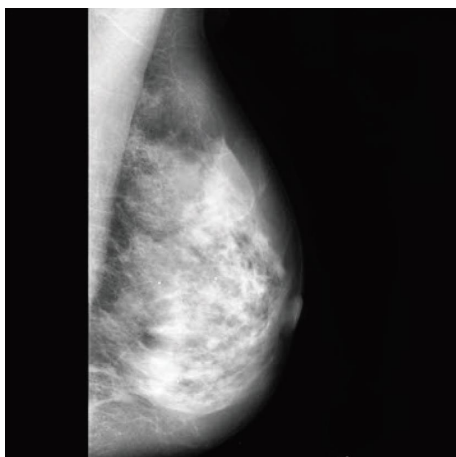


FIGURE 5: Noise reduction with QWTF.

TABLE 3: Comparison of noise reduction approach in this research with previous methods.

References	Noise reduction approach	Windowing size in input image	PSNR (dB)	MSE
Xiao et al. [7]	Median filter	3 × 3	30.69	0.9
		5 × 5	23.94	
		7 × 7	22.51	
Xiao et al. [7]	Mean filter	3 × 3	25.08	0.9
		5 × 5	21.68	
		7 × 7	20.16	
Xiao et al. [7]	Quantum inverse MFT filtering	3 × 3	35.69	1.4
		5 × 5	32.40	
		7 × 7	30.78	
Devakumari and Punithavathi [3]	Adaptive fuzzy median filter	3 × 3	33.60	1.3
		5 × 5	37.15	
		7 × 7	38.39	
Proposed method	QWTF	3 × 3	34.57	0.7
		5 × 5	38.41	
		7 × 7	43.50	



FIGURE 6: Image segmentation with morphology.

### 5. Discussion

Since medical diagnosis systems require reliable and fast methods to ensure doctors, it is essential to use smartification principles in developing such systems.

Moreover, developing smart medical diagnosis systems can reduce human errors and help doctors diagnose diseases. As a result, the early diagnosis will help

TABLE 4: Comparison between the proposed image segmentation method and other methods.

References	Accuracy (%)	Segmentation time (sec)
Abbass et al. [14]	92.78	10 to 60 sec for different images
Pereira et al. [15]	93.54	11.05
Moeskoops and Chen [20]	81	4
Cordeiro et al. [16]	92.50	2
El Adoui et al. [17]	98.50	4
Dalmis et al. [19]	93.30	4
Milletari et al. [41]	82.39	4
Punitha et al. [23]	97.8	1.7
Mouelhi et al. [22]	98	2
Karabatak [27]	98.54	1
Rouhi et al. [24]	96.47	1.2
Proposed method	98.57	0.5

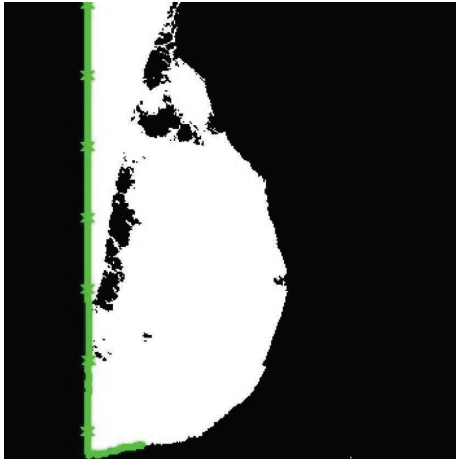


FIGURE 7: Noise reduction after segmentation.



FIGURE 8: The feature extraction output of the pyramid deep CNN.

determine people’s health status and provide them with further care until full recovery. Forming in different areas of the body, cancerous tumors do not have regular

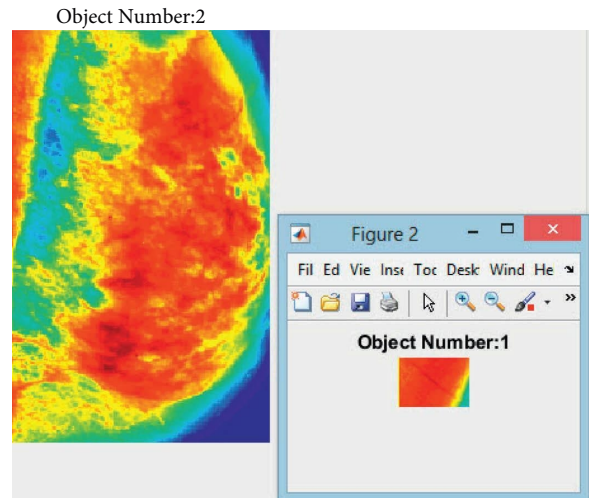


FIGURE 9: An area of a cancerous tumor in the spectral images of the breast through Atrous pyramid CNN.

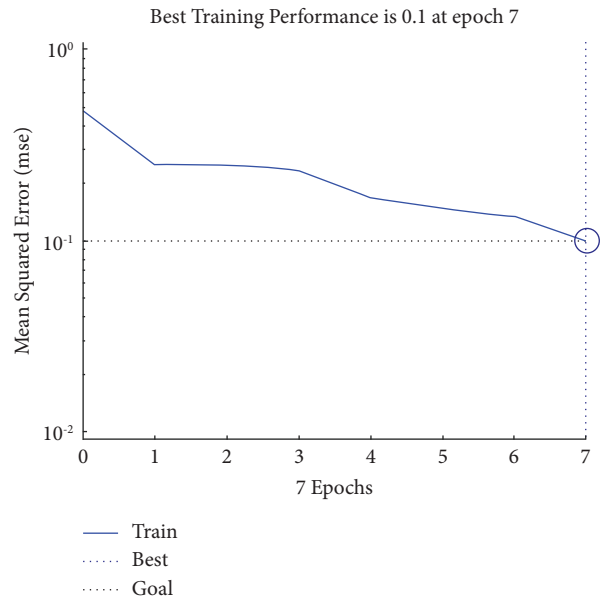


FIGURE 10: Classification efficiency of Atrous pyramid CNN.

shapes and specific patterns. Imaging various areas of the body can help detect cancerous areas and determine the dimensions of these tumors. Medical principles can also be employed to estimate benign and malignant tumors. In fact, it is necessary to diagnose these tumors as accurately as possible, for they are among the most important causes of death all over the world. Thus, smart systems must be developed inevitably. Due to budget and time constraints in this study, we were unable to test the proposed approach on other datasets. Other research constraints included lacking powerful systems for data processing. A totally ordinary system was used in this study. Its specifications were already mentioned.

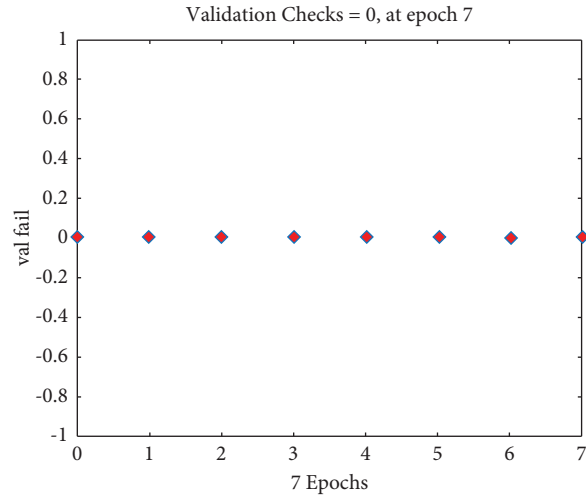


FIGURE 11: Training modes of classification with Atrous pyramid CNN.

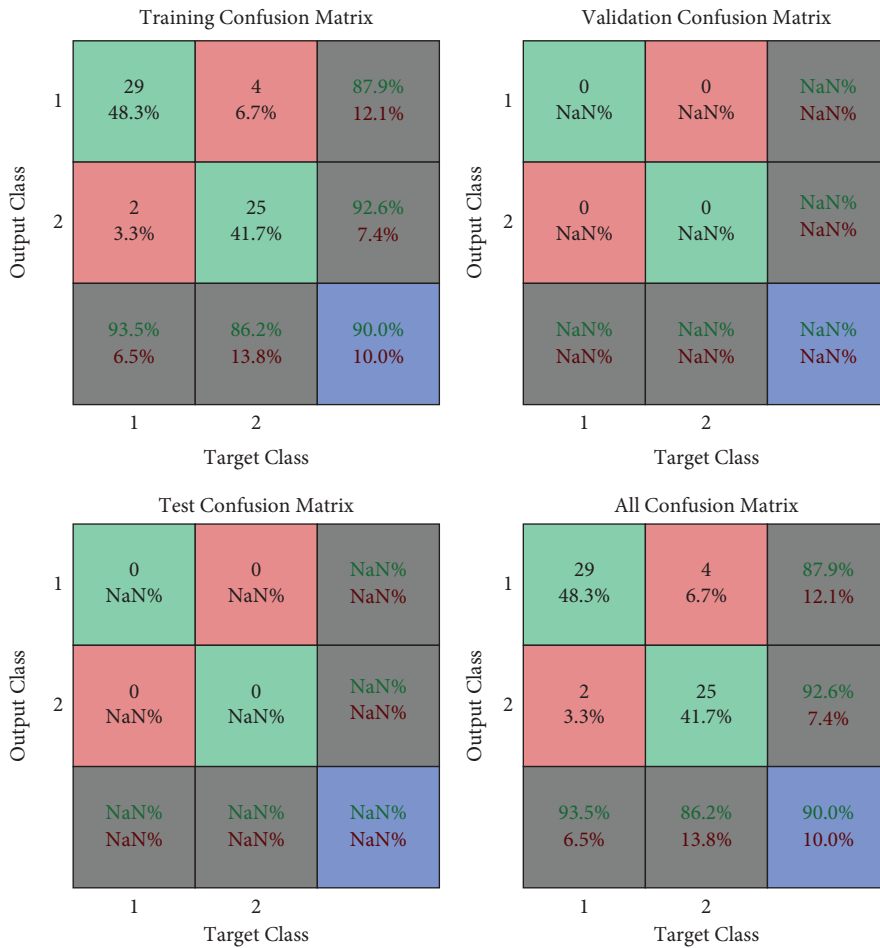


FIGURE 12: Confusion matrix of classification with Atrous pyramid CNN.

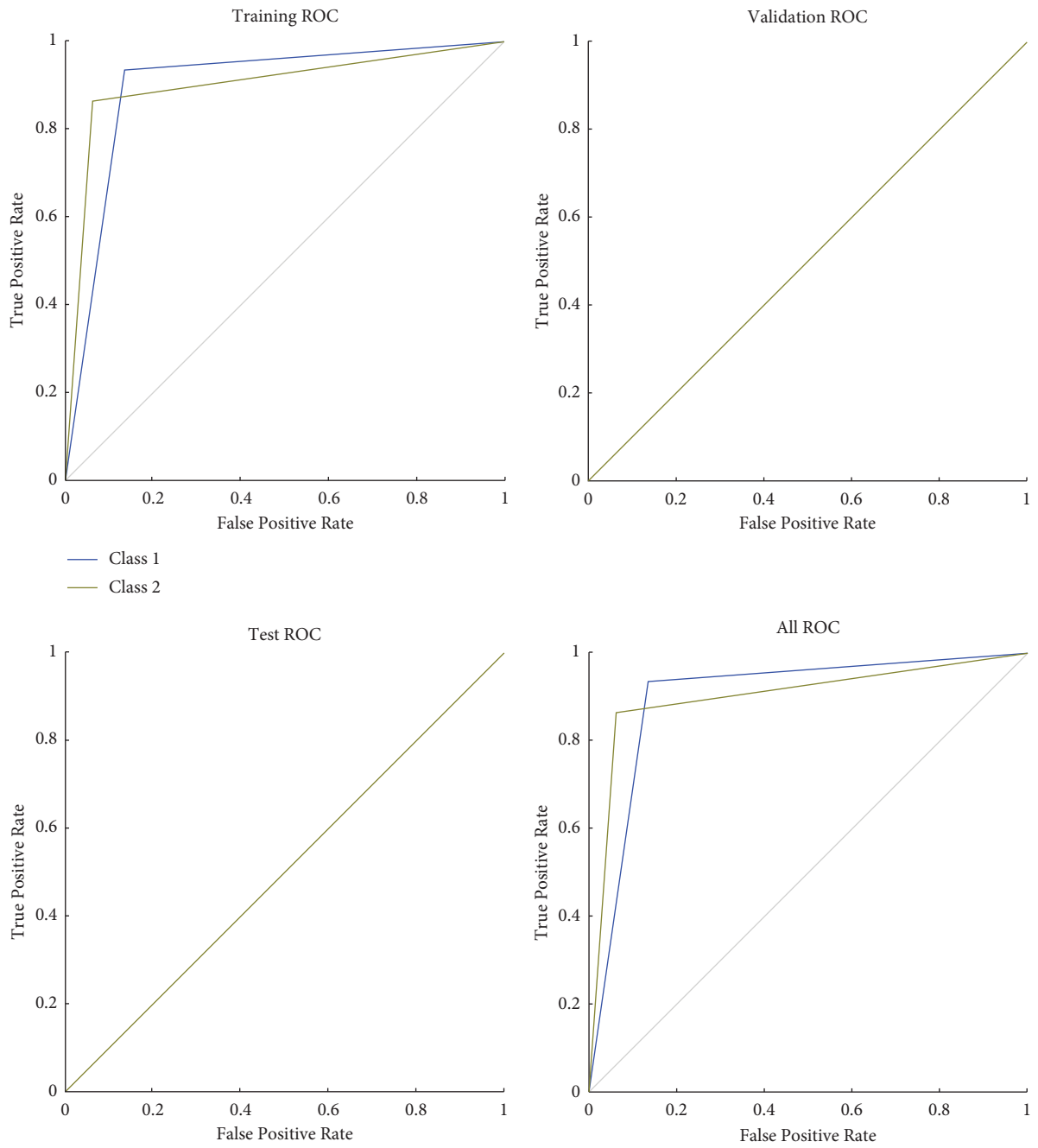


FIGURE 13: ROC of classification with Atrous pyramid CNN.

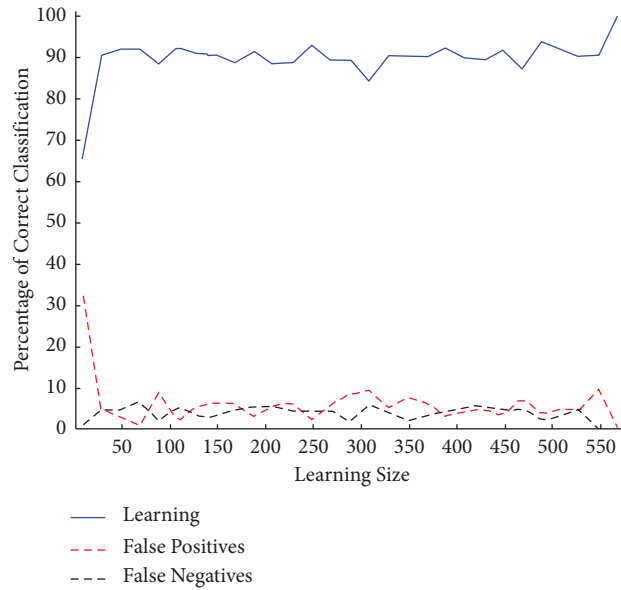


FIGURE 14: Accurate results of classification.

TABLE 5: The results of evaluation criteria for the proposed approach.

AUC	Sensitivity (%)	Feature rate (%)	Precision (%)	MSE	Dice similarity score (DSS) (%)
88.77	92.00	88.00	98.57	0.018	90.00

TABLE 6: The results of comparing the proposed approach with previous methods.

References	Precision (%)
Dehghan Khalilabad and Hassanpour [25]	95.45%
Kaymak et al., [26]	70.4%
Geweid and Abdallah [42]	85%
Karabatak [27]	98.54%
Wang et al. [28]	97.10
Rouhi et al. [24]	96.47%
Proposed method (Atrous pyramid CNN)	98.57%

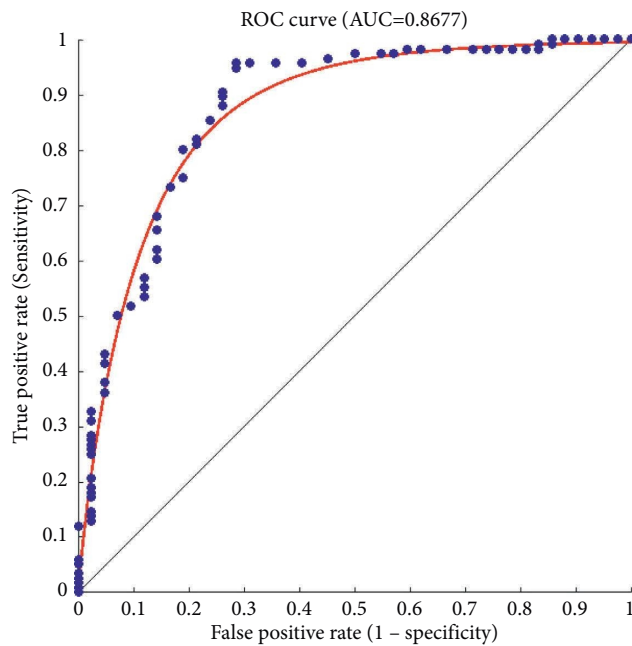


FIGURE 15: AUC and ROC curves for the overall results of the proposed approach.





FIGURE 16: Sample for the detection of cancerous masses at the end of the process.

## 6. Conclusion

The early diagnosis of breast cancer helps prevent the growth of malignant tumors. Thus, it is necessary to develop an intelligent diagnosis model in order to reduce human errors and accelerate cancer diagnosis. This study proposed a novel technique to detect breast cancer on mammographic images and classify benign, malignant, and suspicious tissues. The MIAS dataset consisting of mammographic images and features in breast cancer detection was employed. The proposed model is based on image processing and deep learning. The input system is introduced to the system and preprocessed using the quantum wavelet transform algorithm to reduce noise. Then, morphological image processing is carried out through erosion and dilation operations and boundary extraction to implement segmentation and identify features. Then, image improvement is performed through the quantum wavelet transform algorithm. The features and image are fed as input to the CNN, and windowing is performed through layering. Then, the extracted features and classification are provided. To handle the classification challenges of pyramid CNNs, an Atrous CNN was employed. The proposed approach was found to outperform earlier methodologies in noise reduction and image segmentation. It had also a better receiver operating characteristic (ROC) curve and a larger area under the ROC curve (AUC). The accuracy, sensitivity, specificity, and DSS of the proposed model were obtained to be 98.57%, 92%, 88%, and 90%, respectively. Furthermore, the AUC rate and ROC were calculated to be 88.77%.

## Data Availability

The mini-MIAS database of mammograms is available at <https://peipa.essex.ac.uk/info/mias.html>.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This study was supported by personal financial resources.









## References

- [1] S. Saxena and M. Gyanchandani, "Machine learning methods for computer-aided breast cancer diagnosis using histopathology: a narrative review," *Journal of Medical Imaging and Radiation Sciences*, vol. 51, no. 1, pp. 182–193, 2020.
- [2] H. Feng, J. Cao, H. Wang et al., "A knowledge-driven feature learning and integration method for breast cancer diagnosis on multi-sequence MRI," *Magnetic resonance imaging*, vol. 69, pp. 40–48, 2020.
- [3] D. Devakumar and V. Punithavathi, "Comparison of noise removal filters for breast cancer detection in mammogram images," *International Journal of Pure and Applied Mathematics*, vol. 119, no. 18, pp. 3863–3874, 2018.
- [4] F. Dong, Y. Chen, D.-X. Kong, and B. Yang, "Salt and pepper noise removal based on an approximation of l0 norm," *Computers & Mathematics with Applications*, vol. 70, no. 5, pp. 789–804, 2015.
- [5] A. Ravishankar, S. Anusha, H. Akshatha, A. Raj, S. Jahnavi, and J. Madhura, "A survey on noise reduction techniques in medical images," in *Proceedings of the 2017 International conference of Electronics Communication and Aerospace Technology (ICECA)*, pp. 385–389, IEEE, Beijing China, June 2017.
- [6] T. Kirti, K. Jitendra, and S. Ashok, "Poisson noise reduction from X-ray images by region classification and response median filtering," *Sādhanā*, vol. 42, no. 6, pp. 855–863, 2017.
- [7] S. K. Xiao, N. Subiramaniam, A. Rajamanickam, and A. Balamurugan, "Performance comparison of noise reduction in mammogram images," *International Journal of Renewable Energy Technology*, vol. 05, no. 02, pp. 31–33, 2016.
- [8] X. Jiang, Z. Wang, L. Zhang, and M. Stampanoni, "Noise-analysis-based non-local means method for X-ray grating-based mammography denoising," *IEEE Transactions on Nuclear Science*, vol. 60, no. 2, pp. 802–809, 2013.
- [9] M. Talha, G. B. Sulong, and A. Jaffar, "Preprocessing digital breast mammograms using adaptive weighted frost filter," *Biomedical research*, vol. 27, no. 4, pp. 1407–1412, 2016.
- [10] I. Elyasi, M. A. Pourmina, and M.-S. Moin, "Speckle reduction in breast cancer ultrasound images by using homogeneity modified bayes shrink," *Measurement*, vol. 91, pp. 55–65, 2016.
- [11] D. Eckert, S. Vesal, L. Ritschl, S. Kappler, and A. Maier, "Deep learning-based denoising of mammographic images using physics-driven data augmentation," in *Bildverarbeitung für die Medizin 2020*, pp. 94–100, Springer, Heidelberg, Germany, 2020.
- [12] A. Helwan and R. Abiyev, "Shape and texture features for the identification of breast cancer," *Proceedings of the world congress on engineering and computer science*, vol. 2, pp. 19–21, 2016.
- [13] O. Oladipupo, O. Olajide, S. Olawoye, J. Oyelade, and A. Stephen, "An interval type-2 fuzzy set approach to breast cancer dataset analysis," in *Proceedings of the World Congress on Engineering and Computer Science 2019*, San Francisco, USA, October 2019.

- [14] Q. Abbas, M. E. Celebi, and I. F. García, "Breast mass segmentation using region-based and edge-based methods in a 4-stage multiscale system," *Biomedical Signal Processing and Control*, vol. 8, no. 2, pp. 204–214, 2013.
- [15] D. C. Pereira, R. P. Ramos, and M. Z. Do Nascimento, "Segmentation and detection of breast cancer in mammograms combining wavelet analysis and genetic algorithm," *Computer Methods and Programs in Biomedicine*, vol. 114, no. 1, pp. 88–101, 2014.
- [16] F. R. Cordeiro, W. P. Santos, and A. G. Silva-Filho, "An adaptive semi-supervised Fuzzy GrowCut algorithm to segment masses of regions of interest of mammographic images," *Applied Soft Computing*, vol. 46, pp. 613–628, 2016.
- [17] M. El Adoui, S. A. Mahmoudi, M. A. Larhman, and M. Benjelloun, "MRI breast tumor segmentation using different encoder and decoder CNN architectures," *Computers*, vol. 8, no. 3, p. 52, 2019.
- [18] S. Maqsood, R. Damaševičius, and R. Maskeliūnas, "TTCNN: a breast cancer detection and classification towards computer-aided diagnosis using digital mammography in early stages," *Applied Sciences*, vol. 12, no. 7, p. 3273, 2022.
- [19] M. U. Dalmış, G. Litjens, K. Holland et al., "Using deep learning to segment breast and fibroglandular tissue in MRI volumes," *Medical physics*, vol. 44, no. 2, pp. 533–546, 2017.
- [20] P. Moeskops and L. Chen, "Deep learning for multi-task medical image segmentation in multiple modalities," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 478–486, Springer, NY China, July 2016.
- [21] T. Meraj, W. Alosaimi, B. Alouffi et al., "A quantization assisted U-Net study with ICA and deep features fusion for breast cancer identification using ultrasonic data," *PeerJ Computer Science*, vol. 7, p. e805, 2021.
- [22] A. Mouelhi, H. Rmili, J. B. Ali, M. Sayadi, R. Doghri, and K. Mrad, "Fast unsupervised nuclear segmentation and classification scheme for automatic allred cancer scoring in immunohistochemical breast tissue images," *Computer Methods and Programs in Biomedicine*, vol. 165, pp. 37–51, 2018.
- [23] S. Punitha, A. Amuthan, and K. S. Joseph, "Benign and malignant breast cancer segmentation using optimized region growing technique," *Future Computing and Informatics Journal*, vol. 3, no. 2, pp. 348–358, 2018.
- [24] R. Rouhi, M. Jafari, S. Kasaei, and P. Keshavarzian, "Benign and malignant breast tumors classification based on region growing and CNN segmentation," *Expert Systems with Applications*, vol. 42, no. 3, pp. 990–1002, 2015.
- [25] N. Dehghan Khalilabad and H. Hassanpour, "Employing image processing techniques for cancer detection using microarray images," *Computers in Biology and Medicine*, vol. 81, pp. 139–147, 2017.
- [26] S. Kaymak, A. Helwan, and D. Uzun, "Breast cancer image classification using artificial neural networks," *Procedia Computer Science*, vol. 120, pp. 126–131, 2017.
- [27] M. Karabatak, "A new classifier for breast cancer detection based on Naïve Bayesian," *Measurement*, vol. 72, pp. 32–36, 2015.
- [28] F. Wang, S. Zhang, and L. M. Henderson, "Adaptive decision-making of breast cancer mammography screening: a heuristic-based regression model," *Omega*, vol. 76, pp. 70–84, 2018.
- [29] M. R. Mohebian, H. R. Marateb, M. Mansourian, M. A. Mañanas, and F. Mokarian, "A hybrid computer-aided-diagnosis system for prediction of breast cancer recurrence (HPBCR) using optimized ensemble learning," *Computational and Structural Biotechnology Journal*, vol. 15, pp. 75–85, 2017.
- [30] E. Sadeghipour, N. Sahragard, M.-R. Sayebani, and R. Mahdizadeh, "Breast cancer detection based on a hybrid approach of firefly algorithm and intelligent systems," *Indian Journal of Fundamental and Applied Life Sciences*, vol. 5, p. S1, 2015.
- [31] A. Nomani, Y. Ansari, M. H. Nasirpour, A. Masoumian, E. S. Pour, and A. Valizadeh, "PSOWNNs-CNN: a computational radiology for breast cancer diagnosis improvement based on image processing using machine learning methods," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 5667264, 17 pages, 2022.
- [32] K. Jabeen, M. A. Khan, M. Alhaisoni et al., "Breast cancer classification from ultrasound images using probability-based optimal deep learning feature fusion," *Sensors*, vol. 22, no. 3, p. 807, 2022.
- [33] A. T. Khader, O. A. Alomari, M. A. Al-Betar, and L. M. Abualigah, "Gene selection for cancer classification by combining minimum redundancy maximum relevancy and bat-inspired algorithm," *International Journal of Data Mining and Bioinformatics*, vol. 19, no. 1, pp. 32–51, 2017.
- [34] D. A. Zebari, D. A. Ibrahim, D. Q. Zeebaree et al., "Breast cancer detection using mammogram images with improved multi-fractal dimension approach and feature fusion," *Applied Sciences*, vol. 11, no. 24, Article ID 12122, 2021.
- [35] I. AlShourbaji, P. Kachare, W. Zogaan, L. J. Muhammad, and L. Abualigah, "Learning features using an optimized artificial neural network for breast cancer diagnosis," *SN COMPUT. SCI.*, vol. 3, p. 229, 2022.
- [36] M. Gharaibeh, D. Alzu'Bi, M. Abdullah et al., "Radiology imaging scans for early diagnosis of kidney tumors: a review of data analytics-based machine learning and deep learning approaches," *Big Data and Cognitive Computing*, vol. 6, no. 1, p. 29, 2022.
- [37] E. Sadeghi pour, M. Esmaeili, and M. Romoozi, "Breast cancer diagnosis: a survey of pre- processing, segmentation, feature extraction and classification," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 6, 8708 pages, 2022.
- [38] G. Panelli, B. M. Roberts, and A. Derevianko, "Applying the matched-filter technique to the search for dark matter transients with networks of quantum sensors," *EPJ Quantum Technology*, vol. 7, no. 1, pp. 5–33, 2020.
- [39] P. Mascarade, *Quantum Image Filtering in the Frequency Domain*, Quantum Information and Computation, China, 2016.
- [40] U. P. Shukla and S. J. Nanda, "Parallel social spider clustering algorithm for high dimensional datasets," *Engineering Applications of Artificial Intelligence*, vol. 56, pp. 75–90, 2016.
- [41] F. Milletari, N. Navab, S.-A. Ahmadi, and V-net, "Fully convolutional neural networks for volumetric medical image segmentation," in *Proceedings of the 2016 fourth International Conference on 3D Vision (3DV)*, pp. 565–571, IEEE, Seoul Korea, October 2016.
- [42] G. G. N. Geweid and M. A. Abdallah, "A novel approach for breast cancer investigation and recognition using M-level set-based optimization functions," *IEEE Access*, vol. 7, pp. 136343–136357, 2019.

## Research Article

# Classification of Breast Cancer Histopathological Images Using DenseNet and Transfer Learning

**Musa Adamu Wakili** <sup>1</sup>, **Harisu Abdullahi Shehu** <sup>2</sup>, **Md. Haidar Sharif** <sup>3</sup>,  
**Md. Haris Uddin Sharif** <sup>4</sup>, **Abubakar Umar** <sup>1</sup>, **Huseyin Kusetogullari** <sup>5</sup>,  
**Ibrahim Furkan Ince** <sup>6</sup>, and **Sahin Uyaver** <sup>7</sup>

<sup>1</sup>Abubakar Tafawa Balewa University, Bauchi 740272, Nigeria

<sup>2</sup>School of Engineering and Computer Science, Victoria University of Wellington, Wellington 6012, New Zealand

<sup>3</sup>College of Computer Science and Engineering, University of Hail, Hail 2440, Saudi Arabia

<sup>4</sup>School of Computer & Information Sciences, University of the Cumberland, Williamsburg, KY 40769, USA

<sup>5</sup>Department of Computer Science, Blekinge Institute of Technology, Karlskrona 37141, Sweden

<sup>6</sup>Department of Digital Game Design, Nisantasi University, 34485 Istanbul, Turkey

<sup>7</sup>Department of Energy Science and Technologies, Turkish-German University, 34820 Istanbul, Turkey

Correspondence should be addressed to Md. Haidar Sharif; md.sharif@uoh.edu.sa

Received 9 May 2022; Revised 19 June 2022; Accepted 30 July 2022; Published 10 October 2022

Academic Editor: Mohammed A. A. Al qaness

Copyright © 2022 Musa Adamu Wakili et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Breast cancer is one of the most common invading cancers in women. Analyzing breast cancer is nontrivial and may lead to disagreements among experts. Although deep learning methods achieved an excellent performance in classification tasks including breast cancer histopathological images, the existing state-of-the-art methods are computationally expensive and may overfit due to extracting features from in-distribution images. In this paper, our contribution is mainly twofold. First, we perform a short survey on deep-learning-based models for classifying histopathological images to investigate the most popular and optimized training-testing ratios. Our findings reveal that the most popular training-testing ratio for histopathological image classification is 70%:30%, whereas the best performance (e.g., accuracy) is achieved by using the training-testing ratio of 80%:20% on an identical dataset. Second, we propose a method named DenTnet to classify breast cancer histopathological images chiefly. DenTnet utilizes the principle of transfer learning to solve the problem of extracting features from the same distribution using DenseNet as a backbone model. The proposed DenTnet method is shown to be superior in comparison to a number of leading deep learning methods in terms of detection accuracy (up to 99.28% on BreaKHis dataset deeming training-testing ratio of 80%:20%) with good generalization ability and computational speed. The limitation of existing methods including the requirement of high computation and utilization of the same feature distribution is mitigated by dint of the DenTnet.

## 1. Introduction

Breast cancer is one of the most familiar invasive cancers in women worldwide. Nowadays, it is overtaking lung cancer as the world's chiefly regularly diagnosed cancer [1]. The diagnosis of breast cancer in the early stages significantly decreases the mortality rate by allowing the choice of adequate treatment. With the onset of pattern recognition and machine learning, a good deal of handcrafted or engineered

features-based studies have been proposed for classifying breast cancer histology images. In image classification, feature extraction is a cardinal process used to maximize the classification accuracy by minimizing the number of selected features [2–5]. Deep learning models have the power to automatically extract features, retrieve information, and take in the latest intellectual depictions of data. Thus, they can solve the problems of common feature extraction methods. The automated classification of breast cancer

histopathological images is one of the important tasks in CAD (Computer-Aided Detection/Diagnosis) systems, and deep learning models play a remarkable role by detecting, classifying, and segmenting prime breast cancer histopathological images. Many researchers worldwide have invested appreciable efforts in developing robust computer-aided tools for the classification of breast cancer histopathological images using deep learning. At present, in this research arena, the most popular deep learning models proposed in the literature are based on CNNs [6–66].

A pretrained CNN model, for example, DenseNet [67], utilizes dense connection between layers, reduces the number of parameters, strengthens propagation, and encourages feature reutilization. This improved parameter efficiency makes the network faster and easier to train. Nevertheless, a DenseNet [67] has an excessive connection, as all its layers have a direct connection to each other. Those lavish connections have been shown to decrease the computational and parameter efficiency of the network. In addition, features extracted by a neural network model stay in the same distribution. Therefore, the model might overfit as the features cannot be guaranteed to be sufficient enough. Besides, a CNN-training task demands a large number of training samples; otherwise, it leads to overfitting and reduces generalization ability. However, it is arduous to secure labeled breast cancer histopathological images, which severely limits the classification ability of CNN [27].

On the other hand, the use of transfer learning can expand prior knowledge about data by including information from a different domain to target future data [68]. Consequently, it is a good idea to extract data from a related domain and then transfer those extracted data to the target domain. This way, resources can be saved and the efficiency of the model can be improved during training. A great number of breast cancer diagnosis methods based on transfer learning have been proposed and implemented by distinct researchers (e.g., [57–66]) to achieve state-of-the-art performance (e.g., ACC, AUC, PRS, RES, and F1S) on different datasets. Yet, the limitations of such performance indices, algorithmic assumptions, and computational complexities are indicating a further development of smart algorithms.

In this paper, we aim to propose a novel neural-network-based approach called DenTnet (see Figure 1) for classifying breast cancer histopathological images by taking the benefits of both DenseNet [67] and transfer learning [68]. To address the cross-domain learning problems, we employ the principle of transfer learning for transferring information from a related domain to the target domain. Our proposed DenTnet is anticipated to increase the accuracy of breast cancer histopathological images classification and accelerate the learning process. The DenTnet demonstrates better performance over its alternative CNN and/or transfer-learning-based methods (e.g., see Table 1) on the same dataset as well as training-testing ratio.

To find the best performance scores of deep learning models for classifying histopathological images, contrasting training-testing ratios were applied for divergent models on the same dataset. What would be the most popular and/or

optimized training-testing ratios to classify histopathological images considering existing state-of-the-art deep learning models? There exist many surveys enriched to sufficient contemporary methods and materials with systematic deep discussion of automatic classification of breast cancer histopathological images [68–72]. Nevertheless, to the best of our knowledge, the direct or indirect indication of this question was not reported in any of the previous studies. Henceforth, we perform a succinct survey to investigate this question. Our findings include that the most popular training-testing ratio for histopathological image classification is 70%:30%, whereas the best performance (accuracy) is achieved by using the training-testing ratio of 80%:20% on the identical dataset.

In summary, the main contributions of this context are as follows:

- (i) Determine the most popular and/or optimized training-testing ratios for classifying histopathological images using the existing state-of-the-art deep learning models.
- (ii) Propose a novel approach named DenTnet that amalgamates both DenseNet [67] and transfer learning technique to classify breast cancer histopathological images. DenTnet is anticipated to achieve high accuracy and fasten the learning process due to its utilization of dense connections from its backbone architecture (i.e., DenseNet [67]).
- (iii) Determine the generalization ability of DenTnet and the superiority measure considering nonparametric statistical tests.

The rest of the paper is organized as follows: Section 2 hints some preliminaries; Section 3 surveys briefly the existing deep models for histopathological image classification and reports our findings; Section 4 depicts the architecture of our proposed DenTnet and its implementation details; Section 5 demonstrates the experimental results and comparison on BreaKHis dataset [33]; Section 6 evaluates the generalization ability of DenTnet; Section 7 discusses nonparametric statistical tests, their reported results, and reasons for superiority along with few hints of further study; and Section 8 concludes the paper.

## 2. Preliminaries

Breast cancer is one of the oldest known kinds of cancer first found in Egypt [73]. It is caused by the uncontrolled growth and division of cells in the breast, whereby a mass of tissue called a tumor is created. Nowadays, it is one of the most terrifying cancers in women worldwide. For example, in 2020, there were 2.3 million women diagnosed with breast cancer and 685000 deaths globally [74]. Early detection of breast cancer can save many lives. Breast cancer can be diagnosed in view of histology and radiology images. The radiology images analysis can help to identify the areas, where the abnormality is located. However, they cannot be used to determine whether the area is cancerous [75]. On the other hand, a biopsy is an examination of tissue removed

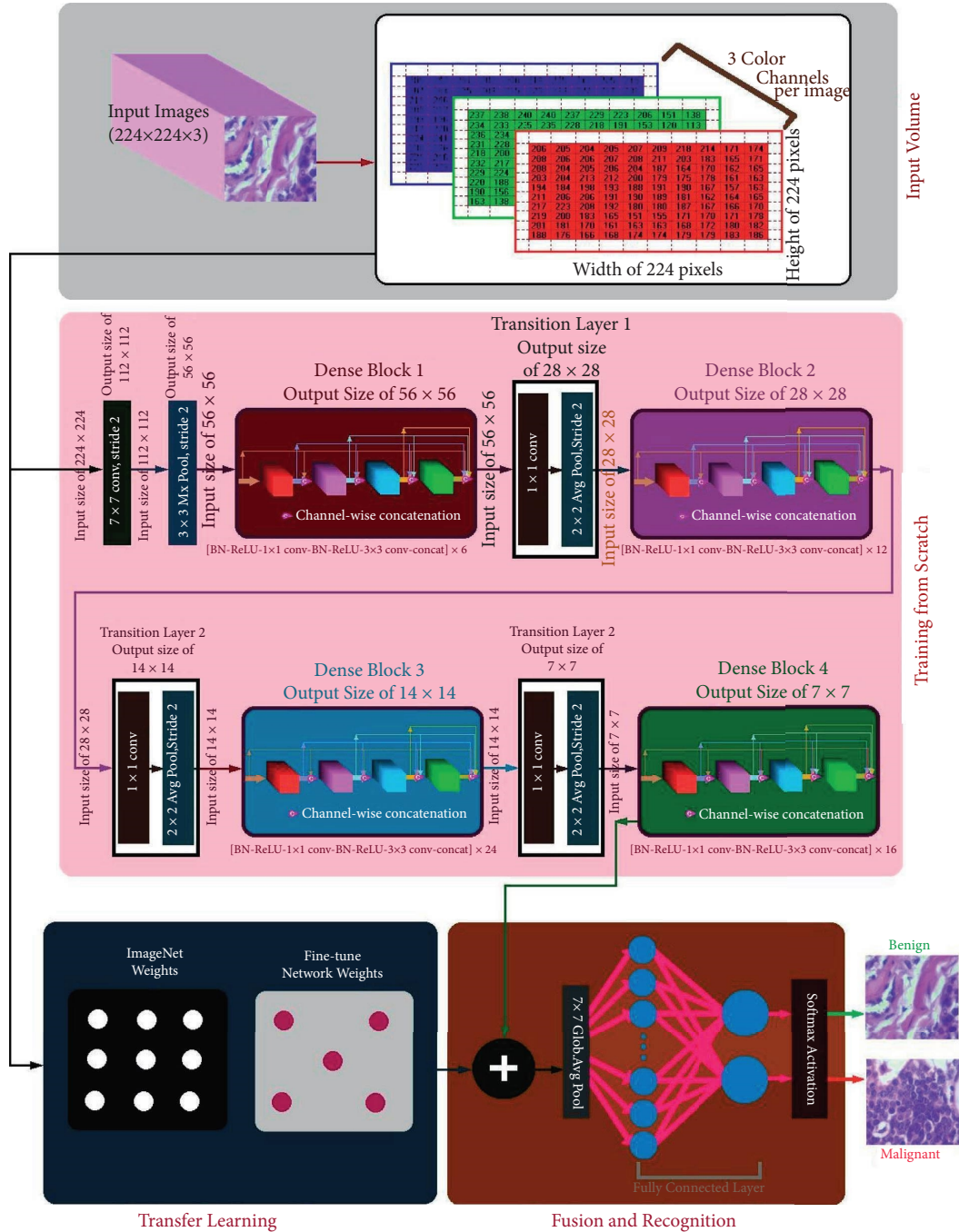


FIGURE 1: Architecture of the proposed DenTnet.

TABLE 1: Comparison of results of various methods using training-testing ratio of 80%: 20% on BreakHis [33]. The best result is shown in bold.

Year	Method	PRS	RES	F1S	AUC	ACC (%)
2020	Togacar et al. [26]	—	—	—	—	97.56
	Parvin et al. [31]	—	—	—	—	91.25
	Man et al. [36]	—	—	—	—	91.44
2021	Boumaraf et al. [63]	—	—	—	—	92.15
	Soumik et al. [60]	—	—	—	—	98.97
2022	Liu et al. [172]	—	—	—	—	96.97
	Zerouaoui and Idri [56]	—	—	—	—	93.85
	Chattopadhyay et al. [174]	—	—	—	—	96.10
	DenTnet [ours]	0.9700	0.9896	0.9948	0.99	<b>99.28</b>

from a living body to discover the presence, cause, or extent of a disease (e.g., cancer). Biopsy is the only reliable way to make sure if an area is cancerous [76]. Upon completion of the biopsy, the diagnosis will be based on the qualification of the histopathologists who determine cancerous regions and malignancy degree [7, 75]. If the histopathologists are not well trained, the histopathology or biopsy report can lead to an incorrect diagnosis. Besides, there might be a lack of specialists, which may cause keeping the tissue samples for up to a few months. In addition, diagnoses made by un-specialized histopathologists are sometimes difficult to replicate. As if that were not enough of a problem, at times, even expert histopathologists tend to disagree with each other. Despite notable progress being reached by diagnostic imaging technologies, the final breast cancer grading and staging are still done by pathologists using visual inspection of histological samples under microscopes.

As analyzing breast cancer is nontrivial and would get down to disagreements among experts, computerized and interdisciplinary systems can improve the accuracy of diagnostic results by reducing the processing time. The CAD can help to assist doctors in reading and interpreting medical images by locating and identifying possible abnormalities in the image [69]. It is proclaimed that the utilization of CAD to automatically classify histopathological images does not only improve the diagnostic efficiency with low cost but also provide doctors with more objective and accurate diagnosis results [77]. Consequently, there is an adamant demand for the CAD [78]. There exist several comprehensive surveys for CAD based methods in the literature. For example, Zebari et al. [71] provided a common description and analysis of existing CAD systems that are utilized in both machine learning and deep learning methods as well as their current state based on mammogram image modalities and classification methods. However, the existing breast cancer diagnosis models take issue with complexity, cost, human-dependency, and inaccuracy [73]. Furthermore, the limitation of datasets is another practical problem in this arena of research. In addition, every deep learning model demands a metric to judge its performance. Explicitly, performance evaluation metrics are the part and parcel of every deep learning model as they indicate progress indices.

In the two following subsections, we discuss the commonly used datasets for classifying histopathological images and the performance evaluation metrics of various deep learning models.

**2.1. Brief Description of Datasets.** Accessing relevant images and datasets is one of the key challenges for image analysis researchers. Datasets and benchmarks enable validating and comparing methods for developing smarter algorithms. Recently, several datasets of breast cancer histopathology images have been released for this purpose. Figure 2 shows a sample breast cancer histopathological image from BreaKHis [33] dataset of a patient who suffered from papillary carcinoma (malignant) with four magnification levels: (a) 40x, (b) 100x (c) 200x, and (d) 400x [79]. The following list of

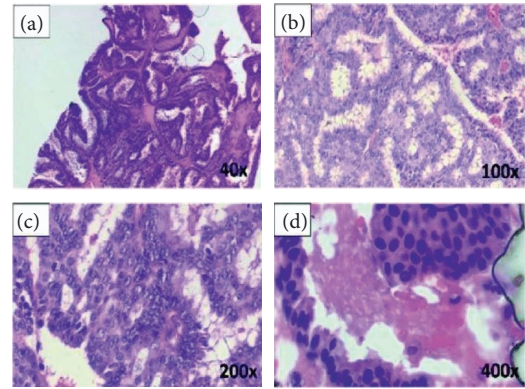


FIGURE 2: A sample breast cancer histopathological image [79] with four magnification levels of (a) 40x, (b) 100x, (c) 200x, and (d) 400x.

datasets has been used in the literature as incorporated in Table 2:

- (i) BreaKHis [33]  $\Rightarrow$  It is considered as the most popular and clinical valued public breast cancer histopathological dataset. It consists of 7909 breast cancer histopathology images, 2480 benign and 5429 malignant samples, from 82 patients with different magnification factors (e.g., 40x, 100x, 200x, and 400x) [33].
- (ii) Bioimaging2015 [122]  $\Rightarrow$  The Bioimaging2015 [122] dataset contained 249 microscopy training images and 36 microscopy testing images in total, equally distributed among the four classes.
- (iii) ICIAR2018 [78]  $\Rightarrow$  This dataset, available as part of the BACH grand challenge [78], was an extended version of the Bioimaging2015 dataset [8, 122]. It contained 100 images in each of four categories (i.e., normal, benign, in situ carcinoma, and invasive carcinoma) [8].
- (iv) BACH [78]  $\Rightarrow$  The database of BACH holds images obtained from ICIAR2018 Grand Challenge [78]. It consists of 400 images with equal distribution of normal (100), benign (100), in situ carcinoma (100), and invasive carcinoma (100). The high-resolution images are digitized with the same conditions and magnification factor of 200x. In this dataset, images have a fixed size of  $2048 \times 1536$  pixels [175].
- (v) TMA [99]  $\Rightarrow$  The TMA (Tissue MicroArray) database from Stanford University is a public resource with an access to 205161 images. All the whole-slide images have been scanned by a 20x magnification factor for the tissue and 40x for the cells [176].
- (vi) Camelyon [97]  $\Rightarrow$  The Camelyon (cancer metastases in lymph nodes) was established based on a research challenge dataset competition in 2016. The Camelyon organizers trained CNNs on smaller datasets for classifying breast cancer in lymph nodes and prostate cancer biopsies. The

TABLE 2: A succinct survey of deep-learning-based histopathological image classification methods. NA indicates either “not available” or “no answer” from the associated authors.

Year	Ref	Aim	Technique	Dataset	Sample	Training (%)	Testing (%)	Result	Performance AUC ACC
2016	Chan and Tuszynski [80]	To predict tumor malignancy in breast cancer	Employed binarization, fractal dimension, SVM	BreakKHis [33]	7909	50	50	ACC of 97.90%, 16.50%, 16.50%, 16.50%, and 25.30% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 39.05%
	Spanhol et al. [33]	To classify histopathological images	Employed CNN based on AlexNet [81]	BreakKHis [33]	7909	70	30	ACC of 90.0%, 88.4%, 84.6%, and 86.1% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively For single-task CNN, ACC of 83.08%, 83.17%, 84.63%, and 82.10%, obtained for 40x, 100x, 200x, and 400x magnification factors, respectively; accordingly, for multitask CNN, ACC of 81.87%, 83.39%, 82.56%, and 80.69%	NA 87.28%
	Bayram-oglu et al. [38]	To classify breast cancer histopathology images	Employed single-task CNN and multitask CNN	BreakKHis [33]	7909	70	30		NA 82.69%
	Abbas [77]	To diagnose breast masses	Applied SURF [82], LBPV [83]	DDSM [84], MIAS [85]	600	40	60	Overall 92%, 84.20%, 91.50%, and 0.91 obtained for sensitivity, specificity, ACC, and AUC, respectively	0.91 91.50%
	Song et al. [21]	To classify histopathology images	Employed a model of CNN, Fisher vector [86], SVM	BreakKHis [33], IICBU2008 [87]	8283	70	30	ACC of 94.42%, 89.49%, 87.25%, and 85.62% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 89.19%
	Wei et al. [22]	To analyze tissue images	Employed a modification of GoogleNet [88]	BreakKHis [33]	7909	75	25	ACC of 97.46%, 97.43%, 97.73%, and 97.74% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 97.59%
	Das et al. [23]	To classify histopathology images	Employed GoogleNet [88]	BreakKHis [33]	7909	80	20	ACC of 94.82%, 94.38%, 94.67%, and 93.49% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 94.34%
	Kahya et al. [89]	To identify features of breast cancer	Employed dimensionality reduction, adaptive sparse SVM	BreakKHis [33]	7909	70	30	ACC of 94.97%, 93.62%, 94.54%, and 94.42% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 94.38%
	Song et al. [24]	To classify histopathology images easily	Employed CNN-based Fisher vector [86], SVM	BreakKHis [33]	7909	70	30	ACC of 90.02%, 88.90%, 86.90%, and 86.30% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 88.03%
	Gupta and Bhavsar [90]	To classify histopathology images	Employed an integrated model	BreakKHis [33]	7909	70	30	Average ACC of 88.09% and 88.40% obtained for image and patient levels, respectively	NA 88.25%
2017	Dhungel et al. [91]	To analyze masses in mammograms	Applied multiscale deep belief nets	INbreast [92]	410	60	20	The best results on the testing set with an ACC got 95% on manual and 91% on the minimal user intervention setup ACC of 84.30%, 84.35%, 85.25% and 82.10% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	0.76 91.03%
	Spanhol et al. [34]	To classify breast cancer images	Using deep CNN	BreakKHis [33]	7900	70	30	ACC of 95.80%, 96.90%, 96.70%, and 94.9% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 96.08%
	Han et al. [35]	To study breast cancer multiclassification	Employed class structure based CNN	BreakKHis [33]	7909	50	50	ACC of 85.75%, 87.03%, and 84.18% obtained for GoogleNet [88], ResNet-50 [75], and CaffeNet [93], respectively	NA 85.65%
	Sun and Binder [39]	To assess performance of H&E stain dat.	A comparative study among ResNet-50 [75], CaffeNet [93], and GoogleNet [88]	BreakKHis [33]	7909	70	30	Overall ACC of 59.0% and 70.4% got from Back-Propagation [95] and Radial Basis [96], respectively	NA 64.70%
	Kaymak et al. [94]	To organize breast cancer images	Back-Propagation [95] and Radial Basis Neural Networks [96]	176 images from a hospital	176	65	35	An AUC of 97.60 (93.60, 100) obtained on par with Camelyon16 [97] test set performance	0.97 95.00%
	Liu et al. [47]	To detect cancer metastases in images	Employed a CNN architecture	Camelyon16 [97]	110	68	32	ACC of 91.28%, 91.45%, 88.57%, and 84.58% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 88.97%
	Zhi et al. [57]	To diagnose breast cancer images	Employed a variation of VGGNet [98]	BreakKHis [33]	7909	80	20	ACC of 83.00% for benign class and 89.00% for malignant class. AUC of malignant was 93.00% and AUC of benign was also 93.00%	0.93 86.00%
	Chang et al. [58]	To solve the limited amount of training data	Employed CNN model from Inception [88] family (e.g., Inception V3)	BreakKHis [33]	4017	70	30		0.93 86.00%

TABLE 2: Continued.

Year	Ref	Aim	Technique	Dataset	Sample	Training (%)	Testing (%)	Result	Performance AUC ACC
	Jannesari et al. [6]	To classify breast cancer images	Employed variations of Inception [88], ResNet [75]	BreakHis [33], 6402 images from TMA [99]	14311	85	15	With ResNets ACC of 99.80%, 98.70%, 94.80%, and 96.40% obtained for four cancer types. Inception V2 with fine-tuning all layers got ACC of 94.10%	0.99 96.34%
	Bardou et al. [7]	To classify breast cancer based on histology images	Employed CNN topology, data augmentation	BreakHis [33]	7909	70	30	ACC of 98.33%, 97.12%, 97.85%, and 96.15% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 97.36%
	Kumar and Rao [9]	To train CNN for using image classification	Employed CNN topology	BreakHis [33]	7909	70	30	ACC of 85.52%, 83.60%, 84.84%, and 82.67% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 84.16%
	Das et al. [11]	To classify breast histopathology images	Employed variation of CNN model	BreakHis [33]	7909	80	20	ACC of 89.52%, 89.06%, 88.84%, and 87.67% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 88.77%
	Nahid et al. [100]	To classify biomedical breast cancer images	Employed Boltzmann machine [101], Tamura et al. features [102]	BreakHis [33]	7909	70	30	ACC of 88.70%, 85.30%, 88.60%, and 88.40% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 87.75%
	Badejo et al. [103]	To classify medical images	Employed local phase quantization, SVM	BreakHis [33]	7909	70	30	ACC of 91.10%, 90.70%, 86.20%, and 84.30% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 88.08%
	Alireza-zadeh et al. [104]	To arrange breast cancer images	Threshold adjacency [105], quadratic analysis [106]	BreakHis [33]	7909	70	30	ACC of 89.16%, 87.38%, 88.46%, and 86.68% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 87.92%
	Du et al. [13]	To distribute breast cancer images	Employed AlexNet [81]	BreakHis [33]	7909	70	30	ACC of 90.69%, 90.46%, 90.64%, and 90.96% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 90.69%
	Gandom-kar et al. [14]	To model CNN for breast cancer image diagnosis	Employed a variation of ResNet [75] (e.g., ResNet152)	BreakHis [33]	7786	70	30	ACC of 98.60%, 97.90%, 98.30%, and 97.60% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 98.10%
2018	Gupta and Bhavsar [15]	To model CNN for breast cancer image diagnosis	Employed DenseNet [67], XGBoost classifier [107]	BreakHis [33]	7909	70	30	ACC of 94.71%, 95.92%, 96.76%, and 89.11% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 94.12%
	Ben-hammou et al. [17]	To study CNN for breast cancer images	Employed Inception V3 [88] module	BreakHis [33]	7909	70	30	ACC of 87.05%, 82.80%, 85.75%, and 82.70% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 84.58%
	Morillo et al. [108]	To label breast cancer images	Employed KAZE features [109]	BreakHis [33]	7909	70	30	ACC of 86.15%, 80.70%, 77.95%, and 72.00% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 97.20%
	Chattoraj and Vishwakarma [110]	To study breast carcinoma images	Zernike moments [111], entropies of Renyi [112], Yager [113]	BreakHis [33]	7909	70	30	ACC of 87.7%, 85.8%, 88.0%, and 84.6% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 96.53%
	Sharma and Mehra [19]	To analyze behavior of magnification independent breast cancer	Employed models of VGGNet [98] and ResNet [75] (e.g., VGG16, VGG19, and ResNet50)	BreakHis [33]	7909	90	10	Pretrained VGG16 with logistic regression classifier showed the best performance with 92.60% ACC, 95.65% AUC, and 95.95% ACC precision score for 90%–10% training-testing data splitting	0.95 94.28%
	Zheng et al. [114]	To study content-based image retrieval	Employed binarization encoding, Hamming distance [115]	BreakHis [33] and others	16309	70	30	ACC of 47.00%, 40.00%, 40.00%, and 37.00% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 41.00%
	Cascianelli et al. [20]	To study features extraction from images	Employed dimensionality reduction using CNN	BreakHis [33]	7909	75	25	ACC of 84.00%, 88.20%, 87.00%, and 80.30% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 84.88%
	Mukkamala et al. [116]	To study deep model for feature extraction	Employed PCANet [117]	BreakHis [33]	7909	80	20	ACC of 96.12%, 97.41%, 90.99%, and 85.85% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 92.59%
	Mahraban Nejad et al. [51]	To retrieve breast cancer images	Employed a variation of VGGNet [98], SVM	BreakHis [33]	7909	98	02	An average ACC of 80.00% was demonstrated from BreakHis [33]	NA 80.00%
	Rakhliln et al. [118]	To analyze breast cancer images	Several deep neural networks and gradient boosted trees classifier	BACH [78]	400	75	25	For 4-class classification task ACC was 87.2% but for 2-class classification ACC was reported to be 93.8%	0.97 90.50%
	Almasni et al. [119]	To detect breast masses	Applied regional deep learning technique	DDSM [84]	600	80	20	Distinguished between benign and malignant lesions with an overall ACC of 97%	0.96 97.00%



TABLE 2: Continued.

Year	Ref	Aim	Technique	Dataset	Sample	Training (%)	Testing (%)	Result	Performance AUC ACC
	Kassani et al. [8]	To use deep learning for binary classification of breast histology images	Usage of VGG19 [98], MobileNet [120], and DenseNet [67]	BreKHis [33], ICIAR2018 [78], PCam [121], Bioimaging2015 [122]	8594	87	13	Multimodal method got better predictions than single classifiers and other algorithms with ACC of 98.13%, 95.00%, 94.64% and 83.10% obtained for BreKHis [33], ICIAR2018 [78], PCam [121], and Bioimaging2015 [122], respectively	NA 92.72%
	Alom et al. [10]	To classify breast cancer from histopathological images	Inception recurrent residual CNN	BreKHis [33], Bioimaging2015 [122]	8158	70	30	From BreKHis [33], ACC of 97.90%, 97.50%, 97.30%, and 97.40%, obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	0.98 97.53%
	Nahid and Kong [12]	To classify histopathological breast images	Employed RGB histogram [123] with CNN	BreKHis [33]	7909	85	15	ACC of 95.00%, 96.60%, 93.500%, and 94.20% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 94.68%
	Jiang et al. [16]	To use CNN for breast cancer histopathological images	Employed CNN, Squeeze-and-Excitation [124] based ResNet [75]	BreKHis [33]	7909	70	30	The achieved accuracy between 98.87% and 99.34% for the binary classification as well as between 90.66% and 93.81% for the multiclass classification	0.99 95.67%
	Sudharshan et al. [18]	To use instance learning for image sorting	Employed CNN-based multiple instance learning algorithm	BreKHis [33]	7909	70	30	ACC of 86.59%, 84.98%, 83.47%, and 82.79% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 84.46%
2019	Gupta and Bhavsar [25]	To segment breast cancer images	Employed ResNet [75] for multilayer feature extraction	BreKHis [33]	7909	70	30	ACC of 88.37%, 90.29%, 90.54%, and 86.11% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 88.82%
	Vo et al. [125]	To extract visual features from training images	Combined weak classifiers into a stronger classifier	BreKHis [33], Bioimaging2015 [122]	8194	87	13	ACC of 95.10%, 96.30%, 96.90%, and 93.80% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 95.56%
	Qi et al. [32]	To organize breast cancer images	Employed a CNN network to complete the classification task	BreKHis [33]	7909	70	30	ACC of 91.48%, 92.20%, 93.01%, and 92.58% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 92.32%
	Talo [41]	To detect and classify diseases in images	DenseNet [67], ResNet [75] (e.g., DenseNet161, ResNet50)	KimiaPath24 [126]	25241	80	20	DenseNet161 pretrained and ResNet50 achieved ACC of 97.89% and 98.87% on grayscale and color images, respectively	NA 98.38%
	Li et al. [127]	To detect invading component in cancer images	Convolutional autoencoder-based contrast pattern mining framework	361 samples of the breast cancer	361	90	10	ACC was taken into account. The overall ACC achieved was 76.00%, whereas 77.70% was presented for FIS	NA 76.00%
	Ragab et al. [44]	To detect breast cancer from images	AlexNet [81] and SVM	DDSM [84], CBIS-DDSM [128]	2781	70	30	The deep CNN presented an ACC of 73.6%, whereas the SVM demonstrated an ACC of 87.2%	0.88 73.60%
	Romero et al. [45]	To study cancer images	A modification of Inception module [88]	HASHI [129]	151465	63	37	From deep learning networks, an overall ACC of 89.00% was demonstrated along with FIS of 90.00%	0.96 89.00%
	Minh et al. [46]	To diagnose breast cancer images	A modification of ResNet [75] and InceptionV3 [88]	BACH [78]	400	70	20	ACC with 95% for 4 types of cancer classes and ACC with 97.5% for two combined groups of cancer	0.97 96.25%

TABLE 2: Continued.

Year	Ref	Aim	Technique	Dataset	Sample	Training (%)	Testing (%)	Result	Performance AUC ACC
	Stamitsas et al. [130]	To visualize a health system for clinicians	Employed region covariance SVM, multiple instance learning [132]	FABCD [133], BreakKHis [33]	7949	70	15	ACC of 91.27% and 92.00% at the patient and image level, respectively	0.98 91.64%
	Togacar et al. [26]	To analyze breast cancer images rapidly	Employed a ResNet [75] architecture with attention modules	BreakKHis [33]	7909	80	20	ACC of 97.99%, 97.84%, 98.51%, and 95.88% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 97.56%
	Asare et al. [134]	To study breast cancer images	Employed self-training and self-paced learning	BreakKHis [33]	7909	70	30	ACC of 93.58%, 91.04%, 93.38%, and 91.00% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 92.25%
	Gour et al. [28]	To diagnose breast cancer tumors images	Employed a modification of ResNet [75]	BreakKHis [33]	7909	70	30	ACC of 90.69%, 91.12%, 95.36%, and 90.24% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	0.91 92.52%
	Li et al. [29]	To grade pathological images	Employed a modification of Xception network [135]	BreakKHis [33], VIAD [136], LSC [137]	8583	60	40	ACC of 95.13%, 95.21%, 94.09%, and 91.42% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 93.96%
	Feng et al. [138]	To allocate breast cancer images	Deep neural-network-based manifold preserving autoencoder [139]	BreakKHis [33]	7909	70	30	ACC of 90.12%, 88.89%, 91.57%, and 90.25% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 90.53%
	Parvin and Mehedi Hasan [31]	To study CNN models for cancer images	LeNet [140], AlexNet [81], VGGNet [98], ResNet [75], Inception V3 [88]	BreakKHis [33]	7909	80	20	ACC of 89.00%, 92.00%, 94.00% and 90.00% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	0.85 91.25%
2020	Carvalho et al. [141]	To classify histological breast images	Entropies of Shannon [142], Renyi [112], Tsallis [143]	BreakKHis [33]	4960	70	30	ACC of 95.40%, 94.70%, 97.60%, and 95.50% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	0.99 95.80%
	Li et al. [144]	To analyze breast cancer images	Employed global covariance pooling module [145]	BreakKHis [33]	7909	70	30	ACC of 96.00%, 96.16%, 98.01%, and 95.97% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 94.93%
	Man et al. [36]	To classify cancer images	Usage of generative adversarial networks, DenseNet [67]	BreakKHis [33]	7909	80	20	ACC of 97.72%, 96.19%, 86.66%, and 85.18% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 91.44%
	Kumar et al. [37]	To classify human breast cancer and canine mammary tumors	Employed a framework based on a variant of VGGNet [98] (e.g., VGGNet16) and SVM	BreakKHis [33] and CMTHis [37]	8261	70	30	For BreakKHis [33], ACC of 95.94%, 96.22%, 98.15%, and 94.41% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively; the same for CMTHis [37], ACC of 94.54%, 97.22%, 92.07%, and 82.84% obtained ACC was taken into account. Estimation of the standard error of mean was approximately 0.81	0.95 96.93%
	Kaushal and Singla [40]	To detect cancerous cells in images.	Employed a CNN model of self-training and self-paced learning	Total 50 images of various patients	50	90	10		NA 93.10%
	Hameed et al. [43]	To use deep learning for classification of breast cancer images	Variants of VGGNet [98] (e.g., fully trained VGG16, fine-tuned VGG16, fully trained VGG19, and fine-tuned VGG19 models)	Breast cancer images: 675 for training and 170 for testing	845	80	20	The ensemble of fine-tuned VGG16 and VGG19 models offered sensitivity of 97.73% for carcinoma class and overall accuracy of 95.29%. It also offered an F1 score of 95.29%	NA 95.29%
	Alantari et al. [48]	To detect breast lesions in digital X-ray mammograms	Adopted three deep CNN models	INbreast [92], DDSM [84]	1010	70	20	In INbreast [92] mean ACC of 89%, 93%, and 95% for CNN, ResNet50, and Inception-ResNet V2, respectively; 95%, 96%, and 98% for DDSM [146]	0.96 94.08%
	Zhang et al. [49]	To classify breast mass	ResNet [75], DenseNet [67], VGGNet [98]	CBIS-DDSM [128], INbreast [92]	3168	70	30	Overall ACC of 90.91% and 87.93% obtained from CBIS-DDSM [128] and INbreast [92], respectively	0.96 89.42%
	Hassan et al. [59]	To classify breast cancer masses	Modification of AlexNet [22] and GoogLeNet [88]	CBIS-DDSM [128], MIAS [85], INbreast [92], etc	600	75	17	With CBIS-DDSM [128] and INbreast [92] databases, the modified GoogLeNet achieved ACC of 98.46% and 92.5%, respectively	0.97 96.98%

TABLE 2: Continued.

Year	Ref	Aim	Technique	Dataset	Sample	Training (%)	Testing (%)	Result	Performance AUC ACC
	Li et al. [147]	To use high-resolution info of images	Multiview attention-guided multiple instance detection network	BreakHis [33], BACH [78], PUJH [148]	12329	70	30	Overall ACC of 94.87%, 91.32%, and 90.45% obtained from BreakHis [33], BACH [78], and PUJH [148], respectively	0.99 92.21%
	Wang et al. [27]	To divide breast cancer images	Employed a model of CNN and CapsNet [149]	BreakHis [33]	7909	70	30	ACC of 92.71%, 94.52%, 94.03%, and 93.54% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 93.70%
	Albeshish et al. [30]	To analyze VGG16 [98]	Employed a variation of VGGNet [98]	BreakHis [33]	7909	90	10	ACC of 96%, 95.10%, and 87% obtained for polynomial SVM, Radial Basis SVM, and k-nearest neighbors, respectively	NA 92.70%
	Kundale et al. [150]	To classify breast cancer from histology images	Employed SURF [82], DSIFT [151], linear coding [152]	BreakHis [33]	7909	70	30	ACC of 93.35%, 93.86%, 93.73%, and 94.00% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 93.74%
	Attallah et al. [153]	To classify breast cancer from histopathological images	Employed several deep learning techniques including autoencoder [139]	BreakHis [33], ICIAR2018 [78]	7909	70	30	For BreakHis [33], ACC of 99.03%, 99.53%, 98.08%, and 97.56% got for 40x, 100x, 200x, and 400x magnification factors, respectively; for ICIAR2018 [78], ACC was 97.93%	NA 98.43%
	Burçak et al. [154]	To classify breast cancer histopathological images	Stochastic [155], Nesterov [156], Adaptive [157], RMSprop [158], AdaDelta [159], Adam [160]	BreakHis [33]	7909	70	30	ACC was taken into account. The overall ACC of 97.00%, 97.00%, 96.00%, and 96.00% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 96.50%
2021	Hirra et al. [161]	To label breast cancer images	Patch-based deep belief network [162]	HASHI [129]	584	52	30	Images from four different data samples achieved an accuracy of 86%	NA 86.00%
	Elmannat et al. [42]	To extract eminent breast cancer image features	A combination of two deep CNNs	BACH [78]	400	60	20	The overall ACC for the subimage classification was 97.29% and for the carcinoma cases the sensitivity achieved was 99.58%	NA 97.29%
	Baker and Abu Qutaish [163]	To segment breast cancer images	Clustering and global thresholding methods	BACH [78]	400	70	30	The maximum ACC obtained from classifiers and neural network using BACH [78] to detect breast cancer	NA 63.66%
	Soumik et al. [60]	To classify breast cancer images	Employed Inception V3 [88]	BreakHis [33]	7909	80	20	ACC of 99.50%, 98.90%, 98.96% and 98.51% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 98.97%
	Brancati et al. [50]	To analyze gigapixel histopathological images	Employed CNN with a compressing path and a learning path	Camelyon16 [164], TUPAC16 [165]	892	68	32	AUC values of 0.698, 0.639, and 0.654 obtained for max-pooling, average pooling, and combined attention maps, respectively	0.66 NA
	Mahmoud et al. [61]	To classify breast cancer images	Employed transfer learning	Mammography images [166]	7500	80	20	Maximum ACC of 97.80% was claimed by using the given dataset [166]. Sensitivity and specificity were estimated	NA 94.45%
	Munien et al. [62]	To classify breast cancer images	Employed EfficientNet [167]	ICIAR2018 [78]	400	85	15	Overall ACC of 98.33% obtained from ICIAR2018 [78]. Sensitivity was also taken into account	NA 98.33%
	Bounarraf et al. [63]	To analyze breast cancer images	Employed ResNet [75] on ImageNet [168] images	BreakHis [33]	7909	80	20	ACC of 94.49%, 93.27%, 91.29%, 89.56% obtained for 40x, 100x, 200x, and 400x magnification factors, respectively	NA 92.15%
	Saber et al. [64]	To detect breast cancer	Employed transfer learning technique	MIAS [85]	322	80	20	Overall ACC, PRS, FIS, and AUC of 98.96%, 97.35%, 97.66%, and 0.995, respectively, got from MIAS [85]	0.995 98.96%

TABLE 2: Continued.

Year	Ref	Aim	Technique	Dataset	Sample	Training (%)	Testing (%)	Result	Performance AUC ACC
2022	Ameh Joseph et al. [169]	To classify breast cancer images	Employed handcrafted features and dense layer	BreakHis [33]	7909	90	10	ACC of 97.87% for 40x, 97.60% for 100x, 96.10% for 200x, and 96.84% for 400x demonstrated from BreakHis [33]	NA 97.08%
	Reshma et al. [52]	To detect breast cancer	Employed probabilistic transition rules with CNN	BreakHis [33]	7909	90	10	ACC, PRS, RES, FIS, and GMN of 89.13%, 86.23%, 81.47%, 85.38%, and 85.17% demonstrated from BreakHis [33]	NA 89.13%
	Huang et al. [53]	To detect nuclei on breast cancer	Employed mask-region-based CNN	H&E images of patients	537	80	20	PRS, RES, and FIS of 91.28%, 87.68%, and 89.44% demonstrated from the used dataset	NA 95.00%
	Chhipa et al. [170]	To learn efficient representations	Employed magnification prior contrastive similarity	BreakHis [33]	7909	70	30	Maximum mean ACC of 97.04% and 97.81% were got from patient and image levels, respectively using BreakHis [33]	NA 97.42%
	Zou et al. [171]	To classify breast cancer images	Employed channel attention module with nondimensionality reduction	BreakHis [33], BACH [78]	8309	90	10	Average ACC, PRS, RES, and FIS of 97.75%, 95.19%, 97.30%, and 96.30% obtained from BreakHis [33], respectively. ACC of 85% got from BACH [78]	NA 91.37%
	Liu et al. [172]	To classify breast cancer images	Employed autoencoder and Siamese framework	BreakHis [33]	7909	80	20	Average ACC, PRS, RES, FIS, and RTM of 96.97%, 96.47%, 99.15%, 97.82%, and 335 seconds obtained from BreakHis [33], respectively	NA 96.97%
	Jayandhi et al. [54]	To diagnose breast cancer	Employed VGG [98] and SVM	MIAS [85]	322	80	20	Maximum ACC of 98.67% obtained from MIAS [85]. Sensitivity and specificity were also calculated	NA 98.67%
	Sharma and Kumar [55]	To classify breast cancer images	Employed Xception [135] and SVM	BreakHis [33]	2000	75	25	Average ACC, PRS, RES, FIS, and AUC of 95.58%, 95%, 95%, 95%, and 0.98 obtained from BreakHis [33], respectively	0.98 95.58%
	Zerouaoui and Idri [56]	To classify breast cancer images	Employed multilayer perceptron, DenseNet201 [67]	BreakHis [33] and others	NA	80	20	ACC of 92.61%, 92%, 93.93%, and 91.73% on four magnification factors of BreakHis [33]	NA 93.85%
	Soltane et al. [65]	To classify breast cancer images	Employed ResNet [75]	323 colored lymphoma images	323	50	50	A total of 27 misclassifications for 323 samples were claimed. PRS, RES, FIS, and Kappa score were estimated	NA 91.6%
	Naik et al. [173]	To analyze breast cancer images	Employed random forest, k-nearest neighbors, SVM	699 whole-slide images	699	80	20	Random forest algorithm achieved better result for classifying benign and malignant images from 190 testing samples	NA 98.2%
	Chattopadhyay et al. [174]	To classify breast cancer images	Employed dense residual dual-shuffle attention network	BreakHis [33]	7909	80	20	Average ACC, PRS, RES, and FIS of 96.10%, 96.03%, 96.08%, and 96.02%, respectively, obtained from four different magnification levels of BreakHis [33]	NA 96.10%
Alruwaili and Gouda [66]	To detect breast cancer	Employed the principle of transfer learning, ResNet [75]	MIAS [85]	322	80	20	Best results for ACC, PRS, RES, FIS, and AUC of 89.5%, 89.5%, 90%, and 89.5% obtained from MIAS [85], respectively	NA 89.5%	

training dataset consists of 270 whole-slide images; among them 160 are normal slides and 110 slides contain metastases [97].

- (vii) PCam [121]  $\Rightarrow$  It is a modified version of the Patch Camelyon (PCam) dataset, which consists of 327680 microscopy images with  $96 \times 96$ -pixel sized patches extracted from the whole-slide images with a binary label hinting the presence of metastatic tissue [8].
- (viii) HASHI [129]  $\Rightarrow$  Each image in the dataset of HASHI (high-throughput adaptive sampling for whole-slide histopathology image analysis) [129] has the size of  $3002 \times 2384$  [161].
- (ix) MIAS [85]  $\Rightarrow$  The Mammographic Image Analysis Society (MIAS) database of digital mammograms [85] contains 322 mammogram images, each of which has a size of  $1024 \times 1024$  pixels with PGM format [59].
- (x) INbreast [92]  $\Rightarrow$  The INbreast database has a total of 410 images collected from 115 cases (i.e., patients) indicating benign, malignant, and normal cases having sizes of  $2560 \times 3328$  or  $3328 \times 4084$  pixels. It contains 36 benign and 76 malignant masses [92].
- (xi) DDSM [84]  $\Rightarrow$  The DDSM [84] dataset was collected by the expert team at the University of South Florida [84]. It contains 2620 scanned film mammography studies. Explicitly, it involves 2620 breast cases (i.e., patients) categorized in 43 different volumes with average size of  $3000 \times 4800$  pixels [48].
- (xii) CBIS-DDSM [128]  $\Rightarrow$  The CBIS-DDSM [128] is an updated version of the DDSM providing easily accessible data and improved region-of-interest segmentation [128, 146]. The CBIS-DDSM dataset comprises 2781 mammograms in the PNG format [49].
- (xiii) CMTHis [37]  $\Rightarrow$  The CMTHis (Canine Mammary Tumor Histopathological Image) [37] dataset comprises 352 images acquired from 44 clinical cases of canine mammary tumors.
- (xiv) FABCD [133]  $\Rightarrow$  The FABCD (Fully Annotated Breast Cancer Database) [133] consists of 21 annotated images of carcinomas and 19 images of benign tissue taken from 21 patients [130].
- (xv) IICBU2008 [87]  $\Rightarrow$  The IICBU2008 (Image Informatics and Computational Biology Unit) malignant lymphoma dataset contains 374 H&E stained microscopy images captured using bright field microscopy [21].
- (xvi) VLAD [136]  $\Rightarrow$  The VLAD (Vector of Locally Aggregated Descriptors) dataset [136] consists of 300 annotated images with resolution of  $1280 \times 960$  [29].
- (xvii) LSC [137]  $\Rightarrow$  The LSC (Lymphoma Subtype Classification) [137] dataset has been prepared by pathologists from different laboratories to create a

real-world type cohort which contains a larger degree of stain and scanning variances [137]. It consists of 374 images with resolution of  $1388 \times 1040$  [29].

- (xviii) KimiaPath24 [126]  $\Rightarrow$  The official KimiaPath24 [126] dataset consists of a total of 23916 images for training and 1325 images for testing. It is publicly available. It shows various body parts with texture patterns [41].

**2.2. Performance Evaluation Metrics.** Performance evaluation of any deep learning model is an important task. An algorithm may give very pleasing results when evaluated using a metric (e.g., *ACC*), but it may give poor results when evaluated against other metrics (e.g., *FIS*) [177]. Usually, we use classification accuracy to measure the performance of deep learning algorithms. But it is not enough to determine the model perfectly. For truly judge any deep learning algorithm, we can use nonidentical types of evaluation metrics including classification *ACC*, *AUC*, *PRS*, *RES*, *FIS*, *RTM*, and *GMN*.

- (i) *ACC*  $\Rightarrow$  It is normally defined in terms of error or inaccuracy [178]. It can be calculated using the following equation:

$$ACC = \frac{(100)(t_n + t_p)}{t_p + t_n + f_p + f_n}, \quad (1)$$

where  $t_n$  is true negative,  $t_p$  is true positive,  $f_p$  is false positive, and  $f_n$  is false negative. Sometimes, *ACC* and the percent correct classification (*PCC*) can be used interchangeably.

- (ii) *PRS*  $\Rightarrow$  Its best value is 100 and the worst value is just 0. It can be formulated using the following equation:

$$PRS = \frac{(100)(t_p)}{t_p + f_p}. \quad (2)$$

- (iii) *RES*  $\Rightarrow$  It should ideally be 100 (the highest) for a good classifier. It can be calculated using the following equation:

$$RES = \frac{(100)(t_p)}{t_p + f_n}. \quad (3)$$

- (iv) *AUC*  $\Rightarrow$  It is one of the most widely used metrics for evaluation [177–179]. The *AUC* of a classifier equals the probability that the classifier ranks a randomly chosen positive sample higher than a randomly chosen negative sample. The *AUC* varies in value from 0 to 1. If the predictions of a model are 100% wrong, then its *AUC* = 0.00; but if its predictions are 100% correct, then its *AUC* = 1.00.
- (v) *FIS*  $\Rightarrow$  It is the harmonic mean between precision and recall. It is also called the F-score or F-measure. It is used in deep learning [177]. It conveys the balance between the precision and the recall. It

also tells us how many instances it classifies correctly. Its highest possible value is 1, which indicates perfect precision and recall. Its lowest possible value is 0, when either the precision or the recall is zero. It can be formulated as

$$F1S = \frac{2}{1/PRS + 1/RES} = \frac{t_p}{t_p + f_p + f_n/2}, \quad (4)$$

where  $PRS$  is the number of correct positive results divided by the number of positive results predicted with the classifier and  $RES$  is the number of correct positive results divided by the number of all relevant samples.

- (vi)  $RTM \Rightarrow$  Estimating the  $RTM$  complexity of algorithms is mandatory for many applications (e.g., embedded real-time systems [180]). The optimization of the  $RTM$  complexity of algorithms in applications is highly expected. The total  $RTM$  can prove to be one of the most important determinative performance factors in many software-intensive systems.
- (vii)  $GMN \Rightarrow$  It indicates the central tendency or typical value of a set of numbers by considering the product of their values instead of using their sum. It can be used to attain a more accurate measure of returns than the mean or arithmetic mean or average. The  $GMN$  for any set of numbers  $x_1, x_2, x_3, \dots, x_m$  can be defined as
 
$$GMN = \left( \prod_{i=1}^m x_i \right)^{1/m} = \sqrt[m]{x_1 x_2 x_3 \dots x_m}. \quad (5)$$
- (viii)  $MCC \Rightarrow$  The Matthews correlation coefficient ( $MCC$ ) is used as a measure of the quality of binary classifications, introduced by biochemist Brian W. Matthews in 1975.
- (ix)  $\kappa \Rightarrow$  The metric of Cohen's kappa ( $\kappa$ ) can be used to evaluate binary classifications.

### 3. A Succinct Survey of State of the Art

This section deals with a summary of existing studies apposite for the classification of breast cancer histopathological images followed by a short discussion and our findings.

**3.1. Summary of Previous Studies.** Table 2 provides a short summary of previous studies carried out to classify breast cancer from images. Experimental results of miscellaneous deep models in the literature on publicly available datasets demonstrated various degrees of accurate cancer prediction scores. However, as  $AUC$  and  $ACC$  are the most important metrics for breast cancer histopathological images classification [49], the experimental results in Table 2 take them into account as the performance indices.

**3.2. Key Techniques and Challenges.** The CNNs can be regarded as a variant of the standard neural networks. Instead of using fully connected hidden layers, the CNNs introduce the structure of a special network, which comprises so-called alternating convolution and pooling layers. They were first introduced for overcoming known problems of fully connected deep neural networks when handling high dimensionality structured inputs, such as images or speech. From Table 2, it is noticeable that CNNs have become state-of-the-art solutions for breast cancer histology images classification. However, there are still challenges even when using the CNN-based approaches to classify pathological breast cancer images [16], as given below:

- (i) Risk of overfitting  $\Rightarrow$  The number of parameters of CNN increases rapidly depending on how large the network is, which may lead to poor learning.
- (ii) Being cost-intensive  $\Rightarrow$  To get a huge number of labeled breast cancer images is very expensive.
- (iii) Huge training data  $\Rightarrow$  CNNs need to be trained using a lot of images, which might not be easy to find considering that collecting real-world data is a tedious and expensive process.
- (iv) Performance degradation  $\Rightarrow$  Various hyperparameters have a significant influence on the performance of the CNN model. The model's parameters need to be tuned properly to achieve a desirable result [75], which usually is not an easy task.
- (v) Employment difficulty  $\Rightarrow$  In the process of training CNN model, it is usually inevitable to rearrange the learning rate parameters to get a better performance. This makes it arduous for the algorithm to use in real-life applications by nonexpert users [181].

Many methods had been proposed in the literature considering the aforementioned challenges. In 2012, AlexNet [81] architecture was introduced for ImageNet Challenge having error rate of 16%. Later various variations of AlexNet [81] with denser network were introduced. Both AlexNet [81] and VGGNet [98] were the pioneering works that demonstrated the potential of deep neural networks [182]. AlexNet was designed by Alex Krizhevsky [81]. It contained 8 layers; the first 5 were convolutional layers, some of them followed by max-pooling layers, and the last 3 were fully connected layers [81]. It was the first large-scale CNN architecture that did well on ImageNet [183] classification. AlexNet [81] was the winner of the ILSVRC [183] classification, the benchmark in 2012. Nevertheless, it was not very deep. SqueezeNet [184] was proposed to create a smaller neural network with fewer parameters that could be easily fit into computer memory and transmitted over a computer network. It achieved AlexNet [81] level accuracy on ImageNet with 50x fewer parameters. It was compressed to less than 0.5 MB (510x smaller than AlexNet [81]) with model compression techniques. The VGG [98] is a deep CNN used to classify images. The VGG19 is a variant of VGG which consists of 19 layers (i.e., 16 convolution layers and 3 fully connected layers, in addition to 5 max-pooling layers and 1 SoftMax layer) [98]. There exist many variants of

VGG [98] (e.g., VGG11, VGG16, VGG19, etc.). VGG19 has 19.6 billion FLOPs (floating point operations per second). VGG [98] is easy to implement but slow to train. Nowadays, many deep-learning-based methods are implemented on influential backbone networks; among them, both DenseNet [67] and ResNet [75] are very popular. Due to the longer path between the input layer and the output layer, the information vanishes before reaching its destination. DenseNet [67] was developed to minimize this effect. The key base element of ResNet [75] is the residual block. DenseNet [67] concentrates on making the deep learning networks move even deeper as well as simultaneously making them well organized to train by applying shorter connections among layers. In short, ResNet [75] adopts summation, whereas DenseNet [67] deals with concatenation. Yet, the dense concatenation of DenseNet [67] creates a challenge of demanding high GPU (Graphics Processing Unit) memory and more training time [182]. On the other hand, the identity shortcut that balances training in ResNet [75] curbs its representation dimensions [182]. Compendiously, there is a dilemma in the alternative between ResNet [75] and DenseNet [67] for many applications in terms of performance and GPU resources [182].

**3.3. Our Findings.** Although various deep learning models in Table 2 often achieved pretty good scores of AUC and ACC, the models demand a large amount of data but breast cancer diagnosis always suffers from a lack of data. To adopt

artificial data is a tentative solution of this issue, but the determination of the best hyperparameters is extremely difficult. Besides efficient deep learning models, the datasets themselves have some limitations, for example, overinterpretation, which cannot be diagnosed using typical evaluation methods based on the ACC of the model. Deep learning models trained on popular datasets (e.g., BreakHis [33]) may suffer from overinterpretation. In overinterpretation, deep learning models make confident predictions based on details that do not make any sense to humans (e.g., promiscuous patterns and image borders). When deep learning models are trained on datasets, they can make apparently authentic predictions based on both meaningful and meaningless subtle signals. This effect, eventually, can reduce the overall classification performance of deep models. Most probably, this is one of the reasons why any state-of-the-art deep learning model in the literature for classifying breast cancer histopathological images (see Table 2) could not show an ACC of 100%.

In addition, the training-testing ratio can regulate the performance of a deep model for image classification. We wish to determine the most popular and/or optimized training-testing ratios for classifying histopathological images using Table 2. To this end, we have calculated the usage frequency of the training-testing ratio (i.e., percentage of the number of papers that used the same ratio) by considering data in Table 2 and the following equation:

$$\text{Usage frequency (\%)} = \frac{(\text{Total number of papers use da training - testing ratio})(100)}{\text{Sum of papers both use da ndu nuse dt hesamet training - testing ratio}}. \quad (6)$$

Figure 3 demonstrates the frequency of usage of training-testing ratio considering data in Table 2. From Figure 3, it is noticeable that the most popular training-testing ratio for histopathological image classification is 70%: 30%. The second-best used training-testing ratio is 80%: 20%, followed by 90%: 10%, 75%: 25%, 50%: 50%, and so on. Figure 4 presents the GMN of ACC for the most frequently used training-testing ratios considering data in Table 2. It shows a different history; in terms of ACC, the rate of 80%: 20% became the best option for the training-testing ratio to classify histopathological images. Explicitly, the GMN of ACC formed like a Gaussian shaped curve and the ratio of 80%: 20% owned its highest peak. To cut a long story short, by considering ACC, the training-testing ratio of 80%: 20% became the finest and the optimal choice for classifying histopathological images.

## 4. Methods and Materials

This section explains in detail our proposed DenTnet model and its implementation. Figure 5 demonstrates a general flowchart of our methodology to classify breast cancer histopathological images automatically.

**4.1. Architecture of Our Proposed DenTnet.** The architecture of our proposed DenTnet is shown in Figure 1, which consists of four different blocks, namely, the input volume, training from scratch, transfer learning, and fusion and recognition.

**4.1.1. Input Volume.** The input is a 3D RGB (three-dimensional red, green, and blue) image with a size of  $224 \times 224$ , that is,  $224 \times 224 \times 3$ .

**4.1.2. Training from Scratch.** Initially, features are extracted from the input images by feeding the input to the convolutional layer. The convolution (conv) layers contain a set of filters (or kernels) parameters, which are learned throughout the training. The size of the filters is usually smaller than the actual image, where each filter convolves with the image and creates an activation map. Thereafter, the pooling layer progressively decreases the spatial size of the representation for reducing the number of parameters in the network. Instead of differentiable functions such as sigmoid and tanh, the network utilizes the ReLU as an activation function. Finally, the extracted features or the output of the last layer from the training from scratch block is then

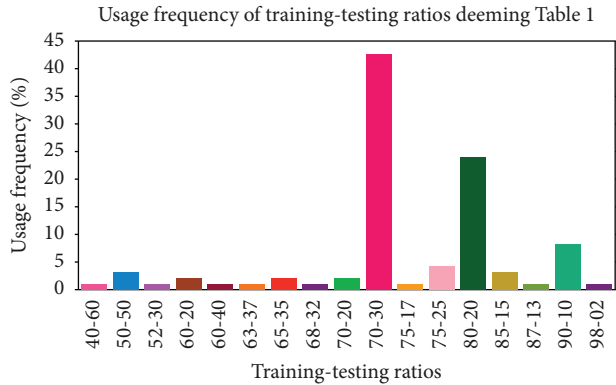


FIGURE 3: Determination of the most popular training-testing ratios using data from Table 2.

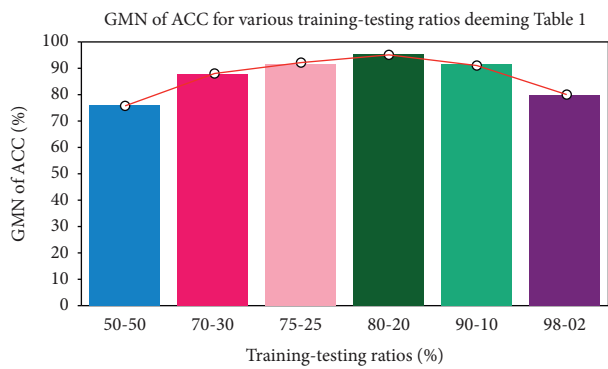


FIGURE 4: GMN of ACC for the most popular training-testing ratios deeming data from Table 2.

amalgamated with the features extracted from the transfer learning approach. Figure 1 includes the design of the DenseNet [67] architecture used to extract the feature using the learning-from-scratch approach.

**4.1.3. Transfer Learning.** In transfer learning, given that a domain  $\mathcal{D}$  consists of feature space  $\mathcal{X}$  and a marginal probability distribution  $P(X)$ , where  $X = x_1, x_2, \dots, x_n \in \mathcal{X}$ , and a task  $\mathcal{T}$  consists of a label space  $\mathcal{Y}$  and an objective predictive function  $f: \mathcal{X} \rightarrow \mathcal{Y}$ , the corresponding label  $f(x)$  of a new instance  $x$  is predicted by function  $f$ , where the new tasks denoted by  $\mathcal{T} = \mathcal{Y}$ ,  $f(x)$  are learned from the training data consisting of pairs  $x_i$  and  $y_i$ , where  $x_i \in \mathcal{X}$  and  $y_i \in \mathcal{Y}$ . When utilizing the learning-from-scratch approach, the extracted features stay in the same distribution. To solve this problem, we amalgamated both learning-from-scratch and the transfer learning approach. The learned parameters are further fine-tuned by retraining the extracted features. This is anticipated to expand the prior knowledge of the network about the data, which might improve the efficiency of the model during training, thereby accelerating the learning speed and also increasing the accuracy of the model. As shown in Figure 1, there is a connection between the blocks of the input volume and transfer learning. The transfer learning approach extracted features from the ImageNet [168] weights. The weight is the parameters (including trainable and nontrainable) learned from the

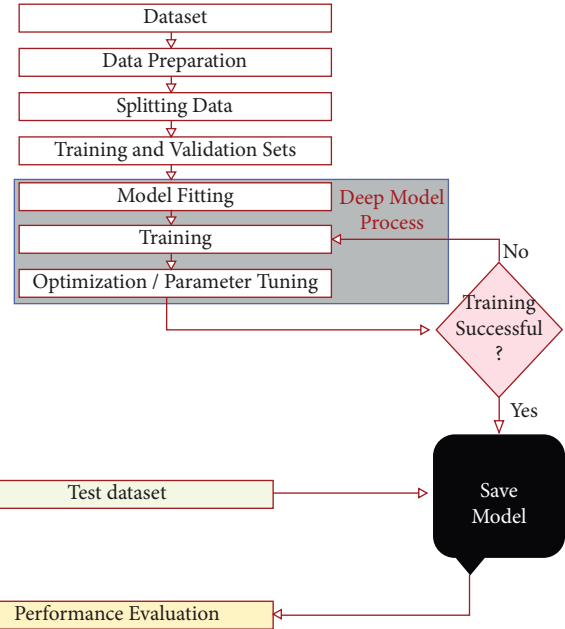


FIGURE 5: Flowchart of our methodology to classify breast cancer histopathological images.

ImageNet [168] dataset. Since transfer learning involves transferring knowledge from one domain to another, we have utilized the ImageNet weight as the models developed in the ImageNet [168] classification competition are measured against each other for performance. Henceforth, the ImageNet weight provides a measure of how good a model is for classification. Besides, the ImageNet weight has already showed a markedly high accuracy [185]. The extracted features are then used by the network before being passed to the fusion and recognition block, where the features are amalgamated with the extracted features from the learning-from-scratch block for recognition.

**4.1.4. Fusion and Recognition.** The extracted features based on the ImageNet weights are then amalgamated with the features extracted by the block of training from scratch. A global average pooling is performed. Dropout technique helps to prevent a model from overfitting. It is used with dense fully connected layers. The fully connected layer compiles the data extracted by previous layers to form the final output. The last step passes the features through the fully connected layer, which then uses SoftMax to classify the class of the input images.

## 4.2. Implementation Details

**4.2.1. Data Preparation.** We have adopted data augmentation, stain normalization, and image normalization strategies to optimize the training process. Hereby, we have explained each of them briefly.

**4.2.2. Data Augmentation.** Due to the limited size of the input samples, the training of our DenTnet was prone to overfitting, which caused low detection rate. One solution to



alleviate this issue was the data augmentation, which generated more training data from the existing training set. Dissimilar data augmentation techniques (e.g., horizontal flipping, rotating, and zooming) were applied to datasets for creating more training samples.

**4.2.3. Stain Normalization.** The breast cancer tissue slices are stained by H&E to differentiate between nuclei stained with purple color and other tissue structures stained with pink and red color to help pathologists analyze the shape of nuclei, density, variability, and overall tissue structure [186]. The H&E staining variability between acquired images exists due to the different staining protocols, scanners, and raw materials. This is a common problem with histological image analysis. Therefore, stain normalization of H&E-stained histology slides was a key step for reducing the color variation and obtaining a better color consistency prior to feeding input images into the DenTnet architecture. Different techniques are available for stain normalization in histological images. We have considered Macenko technique [187] due to its promising performance in many studies to standardize the color intensity of the tissue. This technique was based on a singular value decomposition. A logarithmic function was used to adaptively transform color concentration of the original histopathological image into its optical density (OD) image as  $OD = -\log(I/I_0)$ , where  $OD$  hints the matrix of optical density values,  $I$  belongs to the image intensity in red-green-blue space, and  $I_0$  addresses the illuminating intensity incident on the histological sample.

**4.2.4. Intensity Normalization.** Intensity normalization was another important preprocessing step. Its primary aim was to get the same range of values for each input image before feeding to the DenTnet. It also speeded up the convergence of DenTnet. Input images were normalized to the standard normal distribution by min-max normalization (i.e., using one of the most popular ways to normalize data) to the intensity range of  $[0, 1]$ , which can be computed as

$$x_{normalized} = \frac{x - x_{min}}{x_{max} - x_{min}}, \quad (7)$$

where  $x$ ,  $x_{min}$ , and  $x_{max}$  indicate pixel, minimum, and maximum intensity values of the input image, respectively.

**4.2.5. Hardware and Software Requirements.** DenTnet was implemented using the TensorFlow and Keras framework [188, 189] and coded in Python using Jupyter Notebook on a Kaggle Private Kernel. The experiment was performed on a machine with the following configuration: Intel® Xeon® CPU @ 2.30 GHz with 16 CPU Cores, 16 GB RAM, and NVIDIA Tesla P100 GPU. We implemented and trained everything on the cloud using Kaggle GPU hours.

**4.2.6. Training and Testing Setup.** The dataset was divided in a 80%: 20% ratio, where 80% was used for training and the remaining 20% was used for testing. The data used for testing were kept isolated from the training set and never seen by the

model during training. To evaluate the images classification, we have computed the recognition rate at the image level over the two different classes: (i) correctly classified images and (ii) the total number of images in the test set.

**4.2.7. Training Procedure.** In the training of a neural network, a measure of error is required to compute the error between the targeted output and the computed output of training data known as the loss function. An optimization algorithm is needed to minimize this function. We have considered Adam optimizer [190] with numerical stability constant  $epsilon = None$ ,  $decay = 0.0$ , and  $AMSGrad = True$ . Table 3 presents the hyperparameter values of the proposed deep learning model. Learning rate (also referred to as step size) signifies the proportion to which weights are updated. A smaller value (e.g., 0.000001) slows down the learning process during training, whereas a larger value (e.g., 0.400) results in faster learning. We have considered a learning rate of 0.001. The exponential decay rates of the first and second moments were estimated to be 0.60 and 0.90, respectively. To update the weights, the number of epochs was set to 50 with 3222 steps per epoch and a batch size of 32. For the BreakHis [33] dataset, we had a training sample of 103104 images, with 12288 validation samples and 697 testing samples. The training process used 10-fold cross-validation, where one of the samples was used to validate the data and the remaining 9 samples were used to train the DenTnet model. The fully connected layer used 1024 filters with a dropout rate of 0.50. Finally, the last layer used two filters with a SoftMax layer to classify the image into two classes (e.g., benign and malignant). We have used categorical cross-entropy as the objective function to quantify the difference between two probability distributions. The whole training process took more than 4 hours for the breast cancer tissue images.

## 5. Experimental Results and Comparison on BreakHis Dataset

This section demonstrates the experimental results achieved from classifying the breast cancer histopathology (i.e., BreakHis [33]) images using our proposed DenTnet model.

Figure 6 shows the performance curves obtained during the training of DenTnet using BreakHis [33] dataset. A normalized confusion matrix for the classification of breast cancer test set images is illustrated in Figure 7(a). The main reason for confusion between benign and malignant breast tissues is their similar textures or expression. Henceforth, careful description of texture is required to remove the confusion between the two classes. For binary classification, 5 images only were misclassified, indicating that DenTnet achieved the highest and best ACC of 99.28%. Figures 7(b) and 7(c) demonstrate the ROC curve and precision-recall curve for classification of benign and malignant images from BreakHis [33] dataset, respectively. AUC of 0.99, sensitivity of 97.73%, and specificity 100% have been reported. Table 4 lists the complete classification report of DenTnet. It achieved an ACC of 99.28%.

TABLE 3: List of hyperparameter values for the proposed deep learning model.

Model	Hyperparameters							
	Beta_1	Beta_2	Learning rate	Epoch	Batch size	Epsilon	Decay	AMSGrad
DenTnet	0.60	0.90	0.001	50	32	None	0.0	True

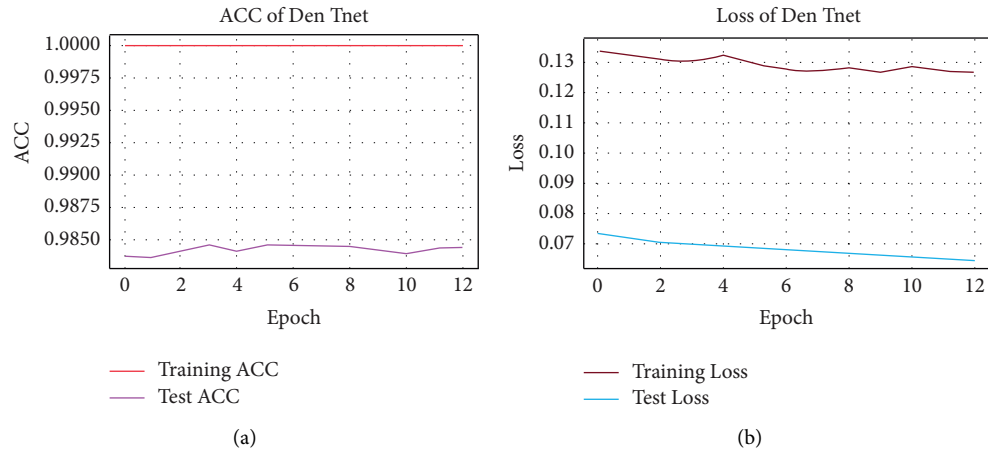


FIGURE 6: (a) hints ACC and (b) shows loss charts of DenTnet during training.

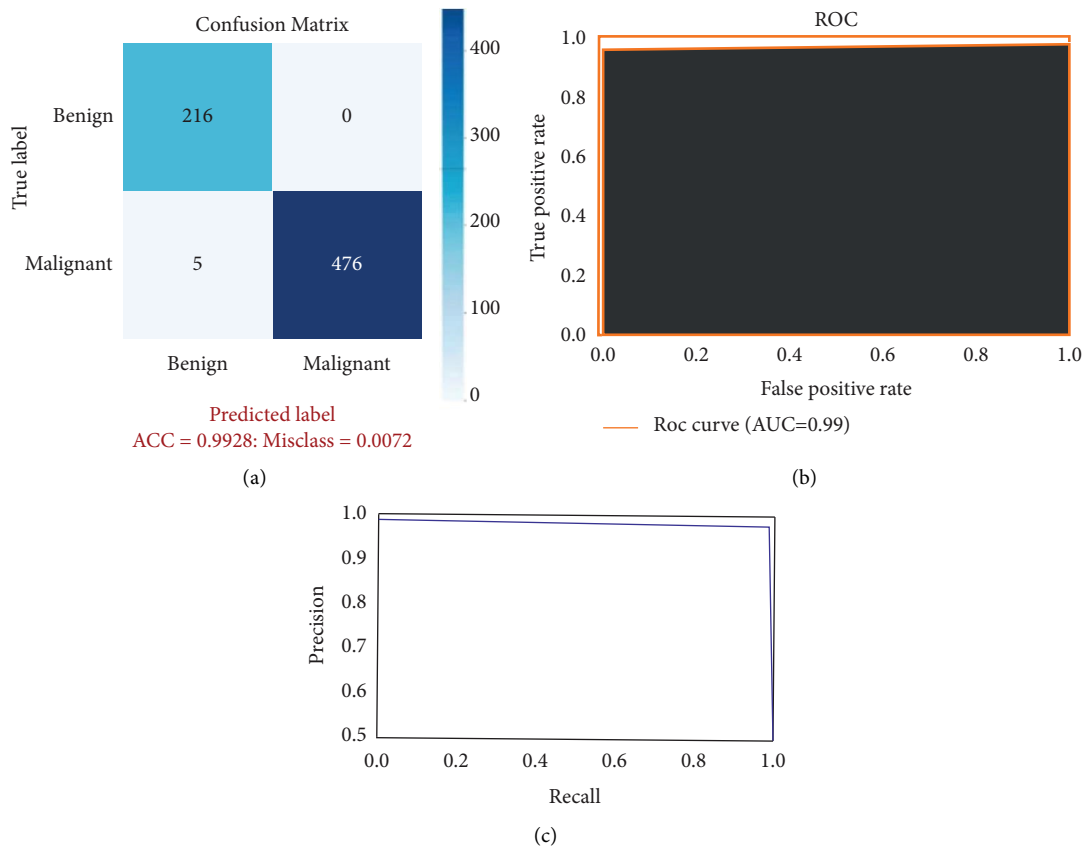


FIGURE 7: (a) hints confusion matrix for benign and malignant classification, (b) shows ROC curve, and (c) demonstrates precision-recall curve.

TABLE 4: Classification results by counting all evaluation criteria.

Type	PRS	RES	F1S	Support
Benign	0.98	1.00	0.99	216
Malignant	1.00	0.99	0.99	481
Micro mean	0.99	0.99	0.99	697
Macro mean	0.99	0.99	0.99	697
Weighted mean	0.99	0.99	0.99	697

Table 1 compares the results obtained by several methods. The methods of Togacar et al. [26], Parvin et al. [31], Man et al. [36], Soumik et al. [60], Liu et al. [172], Zerouaoui and Idri [56], and Chattopadhyay et al. [174] were centered on mainly CNN models, but they were tested against the same training-testing ratio of 80%: 20% on the BreaKHis dataset [33]. However, Boumaraf et al. [63] suggested a transfer-learning-based method deeming the residual CNN ResNet-18 as a backbone model with block-wise fine-tuning strategy and obtained a mean ACC of 92.15% applying a training-testing ratio of 80%: 20% on BreaKHis dataset [33]. From Table 1, it is notable that DenTnet [ours] achieved the best ACC on the same ground.

## 6. Generalization Ability Evaluation of Proposed DenTnet

What would be the performance of the proposed DenTnet compared with other types of cancer or disease datasets? To evaluate the generalization ability of DenTnet, this section presents the experimental result obtained not only from the dataset of BreaKHis [33] but also from additional datasets of Malaria [191], CovidXray [192], and SkinCancer [193].

*6.1. Datasets Irrelevant to Breast Cancer.* The three following datasets are not related to breast cancer. Herewith, their primary aim is to evaluate the generalization ability of our proposed method DenTnet:

- (i) Malaria [191]  $\Rightarrow$  This dataset contains a total of 27558 infected and uninfected images for malaria.
- (ii) SkinCancer [193]  $\Rightarrow$  This dataset contains balanced images from benign skin moles and malignant skin moles. The data consist of two folders, each containing 1800 pictures ( $224 \times 244$ ) from the two types of mole.
- (iii) CovidXray [192]  $\Rightarrow$  Corona (COVID-19) virus affects the respiratory system of healthy individual. The chest X-ray is one of the key imaging methods to identify the coronavirus. This dataset contains chest X-ray of healthy versus pneumonia (Corona) infected patients along with few other categories including SARS (Severe Acute Respiratory Syndrome), *Streptococcus*, and ARDS (Acute Respiratory Distress Syndrome) with a goal of predicting and understanding the infection.

Figure 8 specifies some sample images from Malaria [191], SkinCancer [193], and CovidXray [192] datasets.

*6.2. Experimental Results Comparison.* Using four datasets in the experiment, DenTnet has been compared with six widely used and well-known deep learning models, namely, AlexNet [81], ResNet [75], VGG16 [98], VGG19 [98], Inception V3 [88], and SqueezeNet [184]. To evaluate and analyze the performance of DenTnet, four different cases are considered. The first case is the evaluation of different deep learning methods, which are trained and tested on BreaKHis [33] dataset. The second case studies the performance of the deep-learning-based classification methods that are trained and tested on Malaria [191] dataset. The third case is to train and test the deep learning models on SkinCancer [193] dataset. The final one is to understand and analyze the performance of the deep learning models on CovidXray [192] dataset. The overall results are tabulated in Tables 5–9. Besides, the RTM in seconds of various datasets using the deep learning models is shown in Table 10.

According to the results in terms of GMN of ACC, RES, F1S, and AUC as shown in Tables 5–9, respectively, the proposed DenTnet architecture provides the best scores as compared to AlexNet [81], ResNet [75], VGG16 [98], VGG19 [98], Inception V3 [88], and SqueezeNet [184]. On the other hand, DenTnet gets the third best result. Moreover, in most of the cases, AlexNet [81] obtains the lowest results.

*6.3. Performance Evaluation.* The deepening of deep models makes their parameters rise rapidly, which may lead to overfitting of the model. To take the edge off the overfitting problem, predominantly a large number of dataset images are required as the training set. Considering a small dataset, it is possible to reduce the risk of overfitting of the model by reducing the parameters and augmenting the dataset. Accordingly, DenTnet used fewer parameters along with the dense connections in the construction of the model, instead of the direct connections among the hidden layers of the network. As DenTnet used fewer parameters, it attenuated the vanishing gradient descent and strengthened the feature propagation. Consequently, the proposed DenTnet outperformed its alternative state-of-the-art methods. Yet, its runtime was a bit longer in Malaria [191] and SkinCancer [193] datasets as compared to ResNet [75]. The main reason why the DenTnet model may require more time is that it uses many small convolutions in the network, which can run slower on GPU than compact large convolutions with the same number of GFLOPS. Still, DenTnet includes fewer parameters compatibility when compared to ResNet [75]. Henceforth, it is more efficient in solving the problem of overfitting. In general, all of the used algorithms suffered

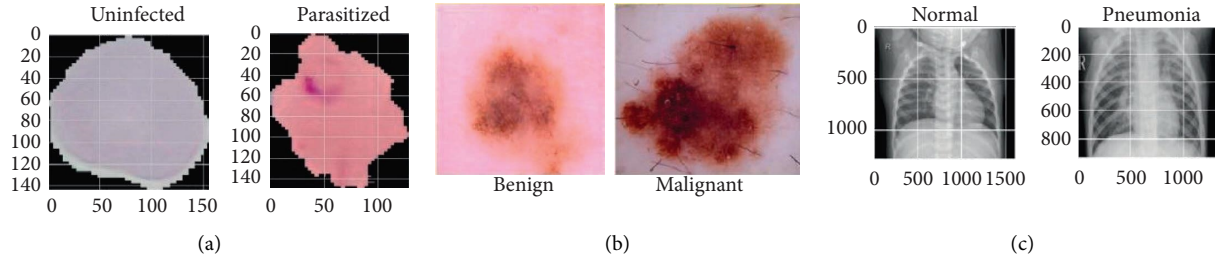


FIGURE 8: (a), (b), and (c) specify images of Malaria [191], SkinCancer [193], and CovidXray [192] datasets, respectively.

TABLE 5: ACC of various methods deeming four different datasets.

Models	ACC of various datasets				GMN of ACC	
	BreaKHis [33]	Malaria [191]	SkinCancer [193]	CovidXray [192]	Success	Failure
AlexNet [81]	0.9268	0.9738	0.8714	0.8526	0.9049	0.0951
ResNet [75]	0.9857	0.9832	0.9045	0.8990	0.9422	0.0578
VGG16 [98]	0.9785	0.9806	0.8501	0.8576	0.9145	0.0855
VGG19 [98]	0.9785	0.9811	0.8512	0.9279	0.9328	0.0672
Inception V3 [88]	0.9784	0.9879	0.8587	0.8998	0.9296	0.0704
SqueezeNet [184]	0.9756	0.9498	0.8288	0.8016	0.8858	0.1142
DenTnet [ours]	0.9928	0.9865	0.9157	0.8942	0.9463	0.0537

TABLE 6: PRS of various methods deeming four different datasets.

Models	PRS of various datasets				GMN of PRS	
	BreaKHis [33]	Malaria [191]	SkinCancer [193]	CovidXray [192]	Success	Failure
AlexNet [81]	0.9317	0.9656	0.8417	0.8744	0.9021	0.0979
ResNet [75]	0.9937	0.9793	0.9167	0.8667	0.9377	0.0623
VGG16 [98]	0.9936	0.9888	0.9055	0.8533	0.9334	0.0666
VGG19 [98]	0.9814	0.9753	0.8083	0.9872	0.9348	0.0652
Inception V3 [88]	0.9829	0.9713	0.8512	0.9796	0.9446	0.0554
SqueezeNet [184]	0.9854	0.9778	0.8871	0.7799	0.9036	0.0964
DenTnet [ours]	0.9700	0.9848	0.9258	0.8641	0.9350	0.0650

TABLE 7: RES of various methods deeming four different datasets.

Models	RES of various datasets				GMN of RES	
	BreaKHis [33]	Malaria [191]	SkinCancer [193]	CovidXray [192]	Success	Failure
AlexNet [81]	0.9647	0.9812	0.9154	0.8880	0.9366	0.0634
ResNet [75]	0.9854	0.9867	0.9010	0.9685	0.9597	0.0403
VGG16 [98]	0.9751	0.9718	0.8250	0.9846	0.9367	0.0633
VGG19 [98]	0.9875	0.9865	0.9065	0.9059	0.9457	0.0543
Inception V3 [88]	0.9854	0.9819	0.8874	0.9491	0.9501	0.0499
SqueezeNet [184]	0.9792	0.9197	0.7861	0.9514	0.9059	0.0941
DenTnet [ours]	0.9896	0.9879	0.9208	0.9629	0.9649	0.0351

TABLE 8: F1S of various methods deeming four different datasets.

Models	F1S of various datasets				GMN of F1S	
	BreaKHis [33]	Malaria [191]	SkinCancer [193]	CovidXray [192]	Success	Failure
AlexNet [81]	0.9479	0.9734	0.8770	0.8811	0.9189	0.0811
ResNet [75]	0.9896	0.9830	0.9129	0.9147	0.9494	0.0506
VGG16 [98]	0.9843	0.9803	0.8634	0.9143	0.9342	0.0658
VGG19 [98]	0.9845	0.9809	0.8546	0.9448	0.9397	0.0603
Inception V3 [88]	0.9844	0.9724	0.8693	0.9077	0.9322	0.0678
SqueezeNet [184]	0.9823	0.9479	0.8336	0.8571	0.9031	0.0969
DenTnet [ours]	0.9948	0.9864	0.9233	0.9108	0.9531	0.0469

TABLE 9: AUC of various methods deeming four different datasets.

Models	AUC of various datasets				GMN of AUC	
	BreaKHis [33]	Malaria [191]	SkinCancer [193]	CovidXray [192]	Success	Failure
AlexNet [81]	0.90	0.97	0.87	0.85	0.8964	0.1036
ResNet [75]	0.99	0.98	0.90	0.91	0.9441	0.0559
VGG16 [98]	0.98	0.98	0.86	0.85	0.9154	0.0846
VGG19 [98]	0.97	0.98	0.85	0.91	0.9260	0.0740
Inception V3 [88]	0.97	0.97	0.89	0.87	0.9239	0.0761
SqueezeNet [184]	0.97	0.95	0.83	0.75	0.8703	0.1297
DenTnet [ours]	0.99	0.99	0.91	0.90	0.9465	0.0535

TABLE 10: RTM of various methods deeming four different datasets.

Models	RTM in seconds of various datasets				GMN of RTM
	BreaKHis [33]	Malaria [191]	SkinCancer [193]	CovidXray [192]	
AlexNet [81]	07573	4100	1413	1328	2762.8
ResNet [75]	16889	3556	0799	2683	3368.5
VGG16 [98]	13419	7698	1450	1081	3567.2
VGG19 [98]	23502	7115	1255	1294	4059.4
Inception V3 [88]	14404	7357	1329	1189	3597.3
SqueezeNet [184]	20080	4140	1339	1864	3795.3
DenTnet [ours]	11083	7102	0873	1519	3196.3

from some degree of overfitting problem on all datasets. We minimized such problems by reducing the batch size and adjusting the learning rate and the dropout rate. In some cases, the proposed DenTnet predicted fewer positive samples as compared to ResNet [75]. This is due to the lack of its conservative designation of the positive class. Thus, the GMN PRS of the proposed DenTnet was about 2% lower than that of ResNet [75].

As VGG16 [98] is easier to implement, many deep learning image classification problems benefit from the technique by using the network either as a sole model or as a backbone architecture to classify images. While VGG19 [98] is better than the VGG16 [98] model, they are both very slow to train—for example, a ResNet with 34 layers only requires 18% of operations as a VGG with 19 layers (around half the layers of the ResNet) will require [194]. Regarding AlexNet [81], the model struggled to scan all features as it is not very deep, resulting in poor performance. The SqueezeNet [184] model achieved approximately the same performance as the AlexNet [81] model. VGG19 [98] and Inception V3 [88] showed almost the same level of effectiveness. Although the ResNet [75] model has proven to be a powerful tool for image classification and is usually fast, it has been shown to take a long time to train. Concisely, using all benefits of DenseNet [67] with optimization, DenTnet obtained the highest GMN ACC of 0.9463, RES of 0.9649, F1S of 0.9531, and AUC of 0.9465 from all four datasets. This implies that DenTnet has the best generalization ability compared to its alternative methods.

Often, it is important to measure that certain deep learning models are more efficient and practical as compared to their alternatives. Seemingly, it is difficult to measure such superiority from the obtained experimental results in Tables 5–10. Nonetheless, nonparametric statistical test can make a clear picture of this issue.

## 7. Nonparametric Statistical Analysis

Figure 9 depicts performance evaluation of various algorithms deeming the numerical values of the ineffectualness metrics and RTM from Table 11. It is noted that, for a better visualization purpose, the RTM scores in Figure 9 use log-normal distribution [195] with a mean of 10 and standard deviation of 1. However, from this graph, it is extremely hard to rank each algorithm. However, statistically, it is possible to show that one algorithm is better than its alternatives. Friedman test [196] and its derivatives (e.g., Iman-Davenport test [197]) are normally referred to as examples of the most well-known nonparametric tests for multiple comparisons. The mathematical equations of Friedman [196], Friedman’s aligned rank [198], and Quade [199] tests can be found in the works of Quade [199] and Westfall and Young [200]. Friedman test [196] takes measures in preparation for ranking of a set of algorithms with performance in descending order. But it can solely inform us about the appearance of differences among all samples of results under comparison. Henceforth, its alternatives (e.g., Friedman’s aligned rank test [198] and Quade test [199]) can give us further information. Consequently, we have performed the tests of Friedman [196], Friedman’s aligned rank [198], and Quade [199] for average rankings based on the features of our experimental study. On rejecting null-hypotheses, we have continued to use post hoc procedures to find the special pairs of algorithms that give idiosyncrasies. In the case of  $1 \times N$  comparisons, the post hoc procedures make up for Bonferroni-Dunn’s [201], Holm’s [202], Hochberg’s [203], Hommel’s [204, 205], Holland and Copenhaver’s [206], Rom’s [207], Finner’s [208], and David Li’s [209] procedures, whereas the post hoc procedures of Nemenyi [210], Shaffer [211], and Bergmann-Hommel [212] are involved in  $N \times N$  comparisons. The details can be found in the works of

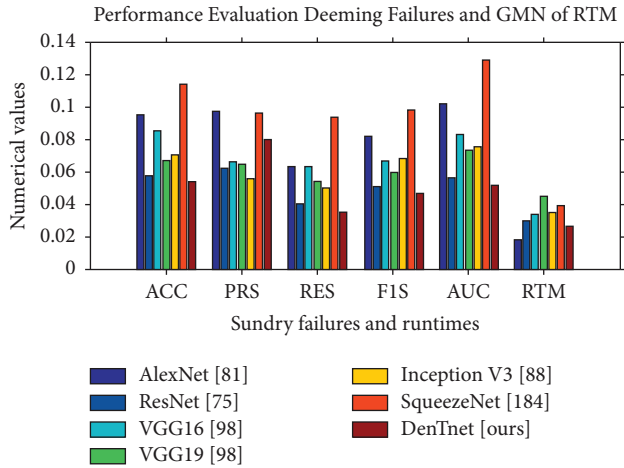


FIGURE 9: Plotting of the numerical values using data from Table 11.

Bergmann and Hommel [212], García and Herrera [213], and Hommel and Bernhard [205].

**7.1. Average Ranking of Algorithms.** To get the nonparametric statistical test results, Friedman [196], Friedman’s aligned rank [198], and Quade [199] tests have been applied to the results of seven models in Table 11. Explicitly, statistical tests have been applied to a matrix with dimension of  $7 \times 6$ , where 7 is the number of models and 6 is the number of parameters (as 6 datasets while applied to the statistical software environment [214]) in each model. Table 12 shows the average ranking computed by using Friedman [196], Friedman’s aligned rank [198], and Quade [199] nonparametric statistical tests. The nonparametric Friedman [196], Friedman’s aligned rank [198], and Quade [199] tests determine whether there were significant differences among various models taking data from Table 11. These tests provide the average ranking of all algorithms; that is, the best performing algorithm gets the highest rank of 1, the second-best algorithm gets the rank of 2, and so on.

Figure 10 makes a visualization of the average rankings using the data in Table 12. From Figure 10, it is noticeable that the algorithm of DenTnet [ours] became the best performing one, with the longest bars of 0.6667, 0.1395, and 0.7242 for Friedman test [196], Friedman’s aligned rank test [198], and Quade test [199], respectively. This indicates that the algorithm of DenTnet [ours] gives great performance for the solution of underlying problems of classifying breast cancer histopathological images from four different datasets. Friedman [196] statistic considered reduction performance (distributed according to chi-square with 6 degrees of freedom) of 24.500000. Friedman’s aligned [198] statistic considered reduction performance (distributed according to chi-square with 6 degrees of freedom) of 23.102557. Iman-Davenport [197] statistic considered reduction performance (distributed according to F-distribution with 6 and 30 degrees of freedom) of 10.652174. Quade [199] statistic considered reduction performance (distributed according to F-distribution with 6 and 30 degrees of freedom) of 5.274194. The  $p$  values computed through Friedman

statistic, Friedman’s aligned statistic, Iman-Davenport statistic, and Quade statistic are 0.000422, 0.000762847204, 0.000002458229, and 0.000820133186, respectively.

Table 13 demonstrates the results obtained on post hoc comparisons of adjusted  $p$  values;  $\alpha = 0.05$  and  $\alpha = 0.10$ . Using level of significance  $\alpha = 0.05$ , (i) Bonferroni-Dunn’s [201] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.008333$ ; (ii) Holm’s [202] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.016667$ ; (iii) Hochberg’s [203] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.0125$ ; (iv) Hommel’s [204] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.016667$ ; (v) Holland’s [206] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.016952$ ; (vi) Rom’s [207] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.013109$ ; (vii) Finner’s [208] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.033617$ ; and (viii) Li’s [209] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.021422$ .

**7.2. Post Hoc Procedures:  $1 \times N$  Comparisons.** In the case of  $1 \times N$  comparisons, the post hoc procedures consist of Bonferroni-Dunn’s [201], Holm’s [202], Hochberg’s [203], Hommel’s [204, 205], Holland and Copenhaver’s [206], Rom’s [207], Finner’s [208], and David Li’s [209] procedures. In these tests, multiple comparison post hoc procedures have been considered for comparing the control algorithm of DenTnet [ours] with others. The results have been shown by computing  $p$  values for each comparison. Table 14 depicts the obtained  $p$  values using the ranks computed by nonparametric Friedman [196], Friedman’s aligned rank [198], and Quade [199] tests. All tests have demonstrated significant improvements of DenTnet [ours] over AlexNet [81], ResNet [75], VGG16 [98], VGG19 [98], Inception V3 [88], and SqueezeNet [184] counting each and every post hoc procedure. Besides, David Li’s [209] procedure had the greatest performance, reaching the lowest  $p$  value in the comparisons.

**7.3. Post Hoc Procedures:  $N \times N$  Comparisons.** In the case of  $N \times N$  comparisons, the post hoc procedures consist of Nemenyi’s [210], Shaffer’s [211], and Bergmann-Hommel’s [212] procedures. Table 15 presents 21 hypotheses of equality among 7 different algorithms and  $p$  values achieved. Using level of significance  $\alpha = 0.05$ , (i) Nemenyi’s [210] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.002381$ ; (ii) Holm’s [202] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.002778$ ; (iii) Shaffer’s [211] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.002381$ ; and (iv) Bergmann’s [212] procedure rejects those hypotheses of AlexNet [81] versus DenTnet [ours], ResNet [75] versus SqueezeNet [184], and SqueezeNet [184] versus DenTnet [ours]. On the other hand, considering  $\alpha = 0.10$ , (i) Nemenyi’s [210] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.004762$ ; (ii) Holm’s [202] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.005556$ ; (iii)

TABLE 11: Summary of performance failure and RTM scores of miscellaneous deep learning algorithms.

Models	GMN scores of performance failure					GMN of RTM
	ACC	PRS	RES	FIS	AUC	
AlexNet [81]	0.0951	0.0979	0.0634	0.0811	0.1036	2762.8
ResNet [75]	0.0578	0.0623	0.0403	0.0506	0.0559	3368.5
VGG16 [98]	0.0855	0.0666	0.0633	0.0658	0.0846	3567.2
VGG19 [98]	0.0672	0.0652	0.0543	0.0603	0.0740	4059.4
Inception V3 [88]	0.0704	0.0554	0.0499	0.0678	0.0761	3597.3
SqueezeNet [184]	0.1142	0.0964	0.0941	0.0969	0.1297	3795.3
DenTnet [ours]	0.0537	0.0650	0.0351	0.0469	0.0535	3196.3

TABLE 12: Average ranking of each algorithm using nonparametric statistical tests. The best results are shown in bold.

Algorithms	Multiple comparison tests		
	Friedman ranking [196]	Friedman’s aligned ranking [198]	Quade ranking [199]
AlexNet [81]	5.3333	26.0000	4.6189
ResNet [75]	2.1667	09.0000	2.2857
VGG16 [98]	4.6667	27.8333	4.6191
VGG19 [98]	4.0000	21.6667	4.3333
Inception V3 [88]	3.6667	22.1667	4.0952
SqueezeNet [184]	6.6667	36.6667	6.6667
DenTnet [ours]	<b>1.5000</b>	<b>07.1667</b>	<b>1.3809</b>
Various statistics	24.500000	23.102557	5.274194
<i>p</i> value	0.000422	0.000763	0.000820

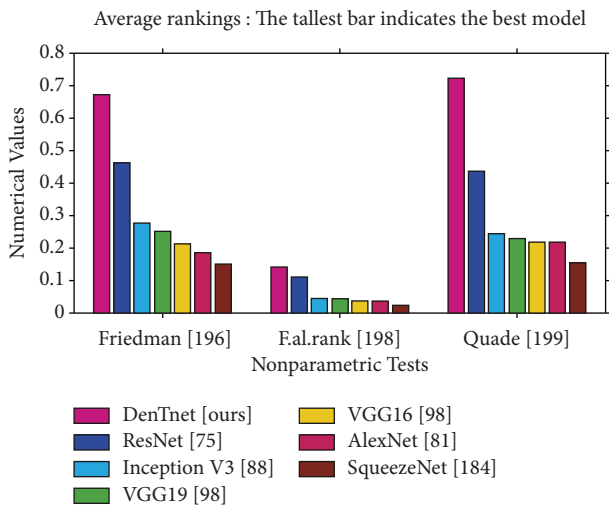


FIGURE 10: Plotting of average rankings data from Table 12, where each value  $x$  is plotted as  $1/x$  to visualize the highest ranking with the tallest bar.

Shaffer’s [211] procedure rejects those hypotheses that have an unadjusted  $p$  value  $\leq 0.004762$ ; and (iv) Bergmann’s [212] procedure rejects those hypotheses of AlexNet [81] versus DenTnet [ours], ResNet [75] versus SqueezeNet [184], and SqueezeNet [184] versus DenTnet [ours].

7.4. *Critical Distance Diagram from Nemenyi [210] Test.* Nemenyi [210] test is very conservative with a low power, and hence it is not a recommended choice in practice [215]. Nevertheless, it has a unique advantage of having an associated plot to demonstrate the results of fair comparison.

Figure 11 depicts the Nemenyi [210] post hoc critical distance diagrams at three distinct levels of significance  $\alpha$  values. If the distance between algorithms is less than the critical distance, then there is no statistically significant difference between them. The diagrams in Figures 11(a) and 11(b) associated with  $\alpha = 0.10$  with the critical distance of 3.3588 and with  $\alpha = 0.05$  with the critical distance of 3.6768, respectively, are identical, whereas the diagram in Figure 11(c) related to  $\alpha = 0.01$  with the critical distance of 4.3054 is different. Any two algorithms are considered as significantly different if their performance variation is greater than the critical distance. To this end, from Figure 11, it is noticeable that, at  $\alpha = 0.01$ , both SqueezeNet [184] versus DenTnet [ours] and SqueezeNet [184] versus ResNet [75] are remarkably different, while other pairs are not remarkably divergent as their performance differences are less than 4.3054. As compared to ResNet [75], DenTnet [ours] differs from SqueezeNet [184] by a greater distance. On the other hand, SqueezeNet [184] versus DenTnet [ours] and AlexNet [81] versus DenTnet [ours] are significantly different at both  $\alpha = 0.10$  and  $\alpha = 0.05$ , whereas SqueezeNet [184] versus ResNet [75] is significantly dissimilar at those  $\alpha$  values. Straightforwardly, DenTnet [ours] is outstandingly unlike both SqueezeNet [184] and AlexNet [81], but ResNet [75] is not outstandingly unlike AlexNet [81]. This implies that the method of DenTnet [ours] outperforms that of ResNet [75], which also agrees with the finding in Figure 10.

7.5. *Reasons of Superiority.* In this study, DenseNet [67] was a great choice as it was very compact and deep. It used less training parameters and reduced the risk of model

TABLE 13: Results achieved on post hoc comparisons for adjusted  $p$  values, with  $\alpha = 0.05$  and  $\alpha = 0.10$ .

Index	Algorithms	$p$ values	$\alpha = 0.05$		$\alpha = 0.10$	
			Holm [202]	Shaffer [211]	Holm [202]	Shaffer [211]
1	VGG19 [98] versus Inception V3 [88]	0.789268	0.050000	0.050000	0.100000	0.100000
2	ResNet [75] versus DenTnet [ours]	0.592980	0.025000	0.025000	0.050000	0.050000
3	VGG16 [98] versus VGG19 [98]	0.592980	0.016667	0.016667	0.033333	0.033333
4	AlexNet [81] versus VGG16 [98]	0.592980	0.012500	0.016667	0.025000	0.033333
5	VGG16 [98] versus Inception V3 [88]	0.422678	0.010000	0.016667	0.020000	0.033333
6	AlexNet [81] versus SqueezeNet [184]	0.285049	0.008333	0.008333	0.016667	0.016667
7	AlexNet [81] versus VGG19 [98]	0.285049	0.007143	0.007143	0.014286	0.014286
8	ResNet [75] versus Inception V3 [88]	0.229102	0.006250	0.006250	0.012500	0.012500
9	AlexNet [81] versus Inception V3 [88]	0.181449	0.005556	0.005556	0.011111	0.011111
10	ResNet [75] versus VGG19 [98]	0.141579	0.005000	0.005000	0.010000	0.010000
11	VGG16 [98] versus SqueezeNet [184]	0.108809	0.004545	0.004545	0.009091	0.009091
12	Inception V3 [88] versus DenTnet [ours]	0.082352	0.004167	0.004167	0.008333	0.008333
13	VGG19 [98] versus DenTnet [ours]	0.045021	0.003846	0.003846	0.007692	0.007692
14	ResNet [75] versus VGG16 [98]	0.045021	0.003571	0.003571	0.007143	0.007143
15	VGG19 [98] versus SqueezeNet [184]	0.032509	0.003333	0.003333	0.006667	0.006667
16	Inception V3 [88] versus SqueezeNet [184]	0.016157	0.003125	0.003333	0.006250	0.006667
17	VGG16 [98] versus DenTnet [ours]	0.011118	0.002941	0.003333	0.005882	0.006667
18	AlexNet [81] versus ResNet [75]	0.011118	0.002778	0.003333	0.005556	0.006667
19	AlexNet [81] versus DenTnet [ours]	0.002116	0.002632	0.003333	0.005263	0.006667
20	ResNet [75] versus SqueezeNet [184]	0.000309	0.002500	0.003333	0.005000	0.006667
21	SqueezeNet [184] versus DenTnet [ours]	0.000034	0.002381	0.002381	0.004762	0.004762

TABLE 14: Adjusted  $p$  values for various tests considering DenTnet [ours] as control method.

Tests	Algorithms	$1 \times N$ post hoc procedures and $p$ values								
		Not adjusted	1-2 step-procedure		Step-down procedures			Step-up procedures		
		$p$ values	$P_{Bonf}$ [201]	$P_{Li}$ [209]	$P_{Holm}$ [202]	$P_{Hol}$ [206]	$P_{Firm}$ [208]	$P_{Hoch}$ [203]	$P_{Hom}$ [204]	$P_{Rom}$ [207]
Friedman [196]	SqueezeNet [184]	0.000034	0.000206	0.000084	0.000206	0.000206	0.000206	0.000206	0.000206	0.000196
	AlexNet [81]	0.002116	0.012694	0.005171	0.010578	0.010533	0.006333	0.010578	0.010578	0.010060
	VGG16 [98]	0.011118	0.066705	0.026588	0.044470	0.043734	0.022112	0.044470	0.044470	0.042403
	VGG19 [98]	0.045021	0.270125	0.099595	0.135063	0.129073	0.066765	0.135063	0.123528	0.135063
	Inception V3 [88]	0.082352	0.494113	0.168281	0.164704	0.157923	0.097990	0.164704	0.164704	0.164704
ResNet [75]	0.592980	3.557881	0.592980	0.592980	0.592980	0.592980	0.592980	0.592980	0.592980	
F. al. rank [198]	SqueezeNet [184]	0.000031	0.000187	0.000152	0.000187	0.000187	0.000187	0.000187	0.000187	0.000178
	VGG16 [98]	0.003525	0.021147	0.016964	0.017623	0.017499	0.010536	0.017623	0.017623	0.016759
	AlexNet [81]	0.007837	0.047023	0.036954	0.031348	0.030982	0.015613	0.031348	0.031348	0.029891
	Inception V3 [88]	0.034193	0.205155	0.143404	0.102578	0.099110	0.050848	0.081277	0.068385	0.081277
	VGG19 [98]	0.040638	0.243830	0.165952	0.102578	0.099110	0.050848	0.081277	0.081277	0.081277
ResNet [75]	0.795758	4.774545	0.795758	0.795758	0.795758	0.795758	0.795758	0.795758	0.795758	
Quade [199]	SqueezeNet [184]	0.027879	0.167272	0.086779	0.167272	0.156038	0.156038	0.167272	0.167272	0.159049
	AlexNet [81]	0.177939	1.067632	0.377531	0.889693	0.624577	0.444463	0.517618	0.388213	0.517618
	VGG16 [98]	0.177939	1.067632	0.377531	0.889693	0.624577	0.444463	0.517618	0.388213	0.517618
	VGG19 [98]	0.219348	1.316086	0.427803	0.889693	0.624577	0.444463	0.517618	0.438695	0.517618
	Inception V3 [88]	0.258809	1.552853	0.468693	0.889693	0.624577	0.444463	0.517618	0.517618	0.517618
ResNet [75]	0.706617	4.239701	0.706617	0.889693	0.706617	0.706617	0.706617	0.706617	0.706617	

overfitting and improved the learning rate. In the dense block of DenTnet, the outputs from the previous layers were concatenated instead of using the summation. This type of concatenation helped to markedly speed up the processing

of data for large number of columns. The dense block of DenTnet contained convolution and nonlinear layers, which applied several optimization techniques (e.g., dropout and BN). DenTnet scaled to hundreds of layers, while exhibiting



TABLE 15: Adjusted  $p$  values of tests for multiple comparisons among all methods.

Index	Hypothesis	$N \times N$ post hoc procedures and $p$ values				
		Unadjusted	Nemenyi [210]	Holm [202]	Shaffer [211]	Bergmann [212]
1	SqueezeNet [184] versus DenTnet [ours]	0.000034	0.000721	0.000721	0.000721	0.000721
2	ResNet [75] versus SqueezeNet [184]	0.000309	0.006479	0.006171	0.004628	0.004628
3	AlexNet [81] versus DenTnet [ours]	0.002116	0.044428	0.040197	0.031734	0.031734
4	AlexNet [81] versus ResNet [75]	0.011118	0.233469	0.200116	0.166763	0.111176
5	VGG16 [98] versus DenTnet [ours]	0.011118	0.233469	0.200116	0.166763	0.122293
6	Inception V3 [88] versus SqueezeNet [184]	0.016157	0.339296	0.258511	0.242354	0.177726
7	VGG19 [98] versus SqueezeNet [184]	0.032509	0.682698	0.487642	0.487642	0.292585
8	ResNet [75] versus VGG16 [98]	0.045021	0.945439	0.630292	0.495230	0.315146
9	VGG19 [98] versus DenTnet [ours]	0.045021	0.945439	0.630292	0.495230	0.405188
10	Inception V3 [88] versus DenTnet [ours]	0.082352	1.729397	0.988227	0.905874	0.494113
11	VGG16 [98] versus SqueezeNet [184]	0.108809	2.284998	1.196904	1.196904	0.652857
12	ResNet [75] versus VGG19 [98]	0.141579	2.973156	1.415789	1.415789	0.652857
13	AlexNet [81] versus Inception V3 [88]	0.181449	3.810433	1.633043	1.633043	1.270144
14	ResNet [75] versus Inception V3 [88]	0.229102	4.811140	1.832815	1.633043	1.270144
15	AlexNet [81] versus VGG19 [98]	0.285049	5.986038	1.995346	1.995346	1.270144
16	AlexNet [81] versus SqueezeNet [184]	0.285049	5.986038	1.995346	1.995346	1.425247
17	VGG16 [98] versus Inception V3 [88]	0.422678	8.876240	2.113390	2.113390	1.690712
18	AlexNet [81] versus VGG16 [98]	0.592980	12.452582	2.371920	2.371920	1.778940
19	VGG16 [98] versus VGG19 [98]	0.592980	12.452582	2.371920	2.371920	1.778940
20	ResNet [75] versus DenTnet [ours]	0.592980	12.452582	2.371920	2.371920	1.778940
21	VGG19 [98] versus Inception V3 [88]	0.789268	16.574629	2.371920	2.371920	1.778940

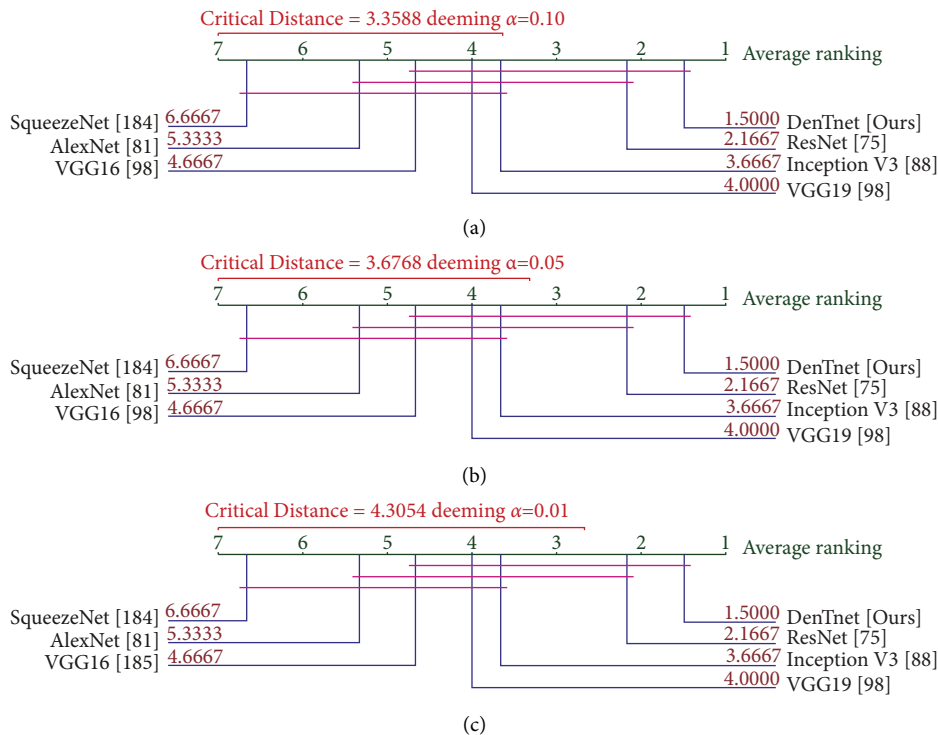


FIGURE 11: Nemenyi [210] post hoc critical distance diagrams for three  $\alpha$  values using data in Table 11.

no optimization difficulties. Overall, this model was applied to a very large number of preprocessed augmented images from BreaKHis [33], Malaria [191], SkinCancer [193], and CovidXray [192] datasets. To the best of our knowledge, no other studies in the literature had such an edge. Additionally, the use of data augmentation approach in this study

positively affected the performance of the model due to expansion in the size of training data, which is the foremost requirement of a deep network for its proper working. Our DenTnet was well trained through various parameters' tuning. For example, in the case of BreaKHis [33], unlike other existing models, our model was trained on all the

magnifications combined (40x, 100x, 200x, and 400x) to avoid any loss of generality.

In sum and substance, based on the aforementioned experimental and nonparametric statistical test results, it is, therefore, possible to conclude that the proposed DenTnet [ours] outperformed AlexNet [81], ResNet [75], VGG16 [98], VGG19 [98], Inception V3 [88], and SqueezeNet [184] in terms of computational speed. Significantly, the accuracy achieved by the proposed DenTnet [ours] surpassed those of existing state-of-the-art models in classifying images of the BreakHis [33], Malaria [191], SkinCancer [193], and the CovidXray [192] dataset.

### 7.6. Limitation of Proposed Model and Methodology.

Despite these promising results, questions remain as to whether the proposed DenTnet model could be utilized to classify categorical images. Moreover, DenTnet was tested with one breast cancer dataset (i.e., BreakHis [33]) only. Although the generalization ability of DenTnet with three non-breast-cancer-related datasets was studied in Section 6, it is unknown whether DenTnet can generalize to other state-of-the-art breast cancer datasets. Future work should, therefore, investigate the efficacy and generalizability of DenTnet with datasets along with multiclass labels, as well as other publicly available breast cancer datasets (e.g., the most recently introduced MITNET dataset [216]).

The classification effect of breast cancer histopathological images of any deep learning methodology is related to the features and many studies predominantly focused on how to develop good feature descriptors and better extract features. Different from traditional handcrafted feature-based models, DenTnet can automatically extract more abstract features. Nevertheless, it is worth noting that although the proposed DenTnet has addressed the cross-domain problem by utilizing the transfer learning approach, features extracted in the methodology are solely deep-network-based features, which are extracted by feeding images directly to the model. However, feeding deep models directly with images would not generalize as the models consider color distribution of an image. It is understood that local information can be captured from color images using Local Binary Pattern (LBP) [217]. Therefore, future work can use multiple types of features by combining the features extracted by the proposed method with LBP features to address this issue.

## 8. Conclusion

We presented that, for classifying breast cancer histopathological images, the most popular training-testing ratio was 70%: 30%, while the best performance was indicated by the training-testing ratio of 80%: 20%. We proposed a novel approach named DenTnet to classify histopathology images using training-testing ratio of 80%: 20%. DenTnet achieved a very high classification accuracy on the BreakHis dataset. Several impediments of existing state-of-the-art methods including the requirement of high computation and the utilization of the identical feature distribution were

attenuated. To test the generalizability of DenTnet, we conducted experiments on three additional datasets (Malaria, SkinCancer, and CovidXray) with varying difficulties. Experimental results on all four datasets demonstrated that DenTnet achieved a better performance in terms of accuracy and computational speed than a large number of effective state-of-the-art classification methods (AlexNet, ResNet, VGG16, VGG19, InceptionV3, and SqueezeNet). These findings contributed to our understanding of how a lightweight model could be used to improve the accuracy and accelerate the learning process of images, including histopathology image classification on using the wild state-of-the-art datasets. Future work shall investigate the efficacy of DenTnet on datasets with multiclass labels.

## Abbreviations

<i>BreaKHis</i> :	Breast cancer histopathological image classification
<i>BACH</i> :	Breast cancer histology images
<i>CNN</i> :	Convolutional neural network
<i>DenseNet</i> :	Dense convolutional Network
<i>ResNet</i> :	Residual network
<i>VGG</i> :	Visual geometry group at the University of Oxford
<i>DDSM</i> :	Digital database for screening mammography
<i>CBIS – DDSM</i> :	Curated breast imaging subset of <i>DDSM</i>
<i>SVM</i> :	Support vector machine
<i>H&amp;E</i> :	Haematoxylin and eosin
<i>BN</i> :	Batch normalization
<i>ReLU</i> :	Rectified linear unit
<i>ROC</i> :	Receiver operating characteristic
<i>AUC</i> :	Area under the ROC curve
<i>ACC</i> :	Accuracy
<i>GMN</i> :	Geometric mean
<i>PRS</i> :	Precision score
<i>RES</i> :	Recall score
<i>F1S</i> :	F1 score
<i>RTM</i> :	Runtime.

## Data Availability

The four following publicly available datasets were used in this study: BreakHis [33] (<https://www.kaggle.com/ambarish/breakhis>), Malaria [191] (<https://www.kaggle.com/iarunava/cell-images-for-detecting-malaria>), CovidXray [192] (<https://github.com/ieee8023/covid-chestxray-dataset>), and SkinCancer [193] (<https://www.kaggle.com/fanconic/skin-cancer-malignant-vs-benign>).

## Conflicts of Interest

The authors have no conflicts of interest to declare.

## References

- [1] Who, *Breast Cancer Now Most Common Form of Cancer: WHO Taking Action*, World Health Organization, Geneva, Switzerland, 2021.

- [2] A. A. Ewees, L. Abualigah, D. Yousri et al., "Improved Slime Mould Algorithm Based on Firefly Algorithm for Feature Selection: A Case Study on QSAR Model," *Engineering with Computers*, vol. 38, pp. 1–15, 2021.
- [3] A. A. Ewees, Z. Y. Algamal, L. Abualigah et al., "A cox proportional-hazards model based on an improved aquila optimizer with whale optimization algorithm operators," *Mathematics*, vol. 10, no. 8, p. 1273, 2022.
- [4] M. AbdElaziz, A. A. Ewees, D. Yousri, L. M. Abualigah, and M. A. A. Al-qaness, "Modified marine predators algorithm for feature selection: case study metabolomics," *Knowledge and Information Systems*, vol. 64, no. 1, pp. 261–287, 2022.
- [5] L. Abualigah, K. H. Almotairi, M. A. Al-qaness et al., "Efficient text document clustering approach using multi-search Arithmetic Optimization Algorithm," *Knowledge-Based Systems*, vol. 248, Article ID 108833, 2022.
- [6] M. Jannesari, M. Habibzadeh, H. Aboulkheyr et al., "Breast cancer histopathological image classification: a deep learning approach," in *Proceedings of the International Conference on Bioinformatics and Biomedicine*, pp. 2405–2412, BIBM, Madrid, Spain, June 2018.
- [7] D. Bardou, K. Zhang, and S. M. Ahmad, "Classification of breast cancer based on histology images using convolutional neural networks," *IEEE Access*, vol. 6, Article ID 24680, 2018.
- [8] S. H. Kassani, P. H. Kassani, M. J. Wesolowski, K. A. Schneider, and R. Deters, "Classification of histopathological biopsy images using ensemble of deep learning networks," in *Proceedings of the Annual International Conference on Computer Science and Software Engineering (CASCON)*, pp. 92–99, Markham, Ontario, July 2019.
- [9] K. Kumar and A. C. S. Rao, "Breast cancer classification of image using convolutional neural network," in *Proceedings of the International Conference on Recent Advances in Information Technology (RAIT)*, pp. 1–6, Dhanbad, India, October 2018.
- [10] M. Z. Alom, C. Yakopcic, M. S. Nasrin, T. M. Taha, and V. K. Asari, "Breast cancer classification from histopathological images with inception recurrent residual convolutional neural network," *Journal of Digital Imaging*, vol. 32, no. 4, pp. 605–617, 2019.
- [11] K. Das, S. Conjeti, A. G. Roy, J. Chatterjee, and D. Sheet, "Multiple instance learning of deep convolutional neural networks for breast histopathology whole slide classification," in *Proceedings of the Int. Symposium on Biomedical Imaging (ISBI)*, pp. 578–581, New York, NY, USA, July 2018.
- [12] A. A. Nahid and Y. Kong, "Histopathological breast-image classification using concatenated R–G–B histogram information," *Annals of Data Science*, vol. 6, no. 3, pp. 513–529, 2019.
- [13] B. Du, Q. Qi, H. Zheng, Y. Huang, and X. Ding, "Breast cancer histopathological image classification via deep active learning and confidence boosting. Artificial neural networks and machine learning (ICANN)," in *Proceedings of the 27th international conference on artificial neural networks*, vol. 11140, pp. 109–116, Greece, May 2018.
- [14] Z. Gandomkar, P. C. Brennan, and C. Mello-Thoms, "MuDeRN: multi-category classification of breast histopathological image using deep residual networks," *Artificial Intelligence in Medicine*, vol. 88, pp. 14–24, 2018.
- [15] V. Gupta and A. Bhavsar, "Sequential modeling of deep features for breast cancer histopathological image classification," in *Proceedings of the Conference on Computer Vision and Pattern Recognition Workshops*, pp. 2254–2261, Salt Lake City, UT, USA, June 2018.
- [16] Y. Jiang, L. Chen, H. Zhang, and X. Xiao, "Breast cancer histopathological image classification using convolutional neural networks with small SE-ResNet module," *PLoS One*, vol. 14, no. 3, Article ID 0214587, 2019.
- [17] Y. Benhammou, S. Tabik, B. Achchab, and F. Herrera, "A first study exploring the performance of the state-of-the-art CNN model in the problem of breast cancer," in *Proceedings of the International Conference on Learning and Optimization Algorithms: Theory and Applications (LOPAL)*, no. 1–47, p. 47, Rabat, Morocco, June 2018.
- [18] P. J. Sudharshan, C. Petitjean, F. A. Spanhol, L. E. Oliveira, L. Heutte, and P. Honeine, "Multiple instance learning for histopathological breast cancer image classification," *Expert Systems with Applications*, vol. 117, pp. 103–111, 2019.
- [19] S. Sharma and R. Mehra, "Breast cancer histology images classification: training from scratch or transfer learning?" *ICT Express*, vol. 4, pp. 247–254, 2018.
- [20] S. Cascianelli, R. Bello-Cerezo, F. Bianconi et al., "Dimensionality reduction strategies for CNN-based classification of histopathological images," *Intelligent Interactive Multimedia Systems and Services*, pp. 21–30, Springer International Publishing: Cham, New York, NY, USA, 2018.
- [21] Y. Song, H. Chang, H. Huang, and W. Cai, "Supervised intra-embedding of Fisher vectors for histopathology image classification," in *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pp. 99–106, Quebec City, Canada, March 2017.
- [22] B. Wei, Z. Han, X. He, and Y. Yin, "Deep learning model based breast cancer histopathological image classification," in *Proceedings of the International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, pp. 348–353, Chengdu, April 2017.
- [23] K. Das, S. P. K. Karri, A. G. Roy, J. Chatterjee, and D. Sheet, "Classifying histopathology whole-slides using fusion of decisions from deep convolutional network on a collection of random multi-views at multi-magnification," in *Proceedings of the International Symposium on Biomedical Imaging (ISBI)*, pp. 1024–1027, Melbourne, Australia, March 2017.
- [24] Y. Song, J. J. Zou, H. Chang, and W. Cai, "Adapting Fisher vectors for histopathology image classification," in *Proceedings of the International Symposium on Biomedical Imaging (ISBI)*, pp. 600–603, Melbourne, Australia, April 2017.
- [25] V. Gupta and A. Bhavsar, "Partially-Independent framework for breast cancer histopathological image classification," in *Proceedings of the Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1123–1130, Long Beach, CA, USA, June 2019.
- [26] M. Togacar, K. B. Ozkurt, B. Ergen, and Z. Comert, "BreastNet: a novel convolutional neural network model through histopathological images for the diagnosis of breast cancer," *Physica A: Statistical Mechanics and Its Applications*, vol. 545, Article ID 123592, 2020.
- [27] P. Wang, J. Wang, Y. Li, P. Li, L. Li, and M. Jiang, "Automatic classification of breast cancer histopathological images based on deep feature fusion and enhanced routing," *Biomedical Signal Processing and Control*, vol. 65, Article ID 102341, 2021.
- [28] M. Gour, S. Jain, and T. Sunil Kumar, "Residual learning based CNN for breast cancer histopathological image classification," *International Journal of Imaging Systems and Technology*, vol. 30, no. 3, pp. 621–635, 2020.

- [29] L. Li, X. Pan, H. Yang et al., "Multi-task deep learning for fine-grained classification and grading in breast cancer histopathological images," *Multimedia Tools and Applications*, vol. 79, no. 21-22, pp. 14509-14528, 2020.
- [30] D. Albashish, R. Al-Sayyed, A. Abdullah, M. H. Ryalat, and N. Ahmad Almansour, "Deep CNN Model Based on VGG16 for Breast Cancer Classification," in *Proceedings of the International Conference on Information Technology (ICIT)*, pp. 805-810, Amman, Jordan, July 2021.
- [31] F. Parvin and M. A. Mehedi Hasan, "A comparative study of different types of convolutional neural networks for breast cancer histopathological image classification," in *Proceedings of the 2020 IEEE Region 10 Symposium (TENSymp)*, pp. 945-948, Dhaka, Bangladesh, June 2020.
- [32] Q. Qi, Y. Li, J. Wang et al., "Label-efficient breast cancer histopathological image classification," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 5, pp. 2108-2116, 2019.
- [33] F. A. Spanhol, L. S. Oliveira, C. Petitjean, and L. Heutte, "Breast cancer histopathological image classification using Convolutional Neural Networks," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pp. 2560-2567, Vancouver, BC, 2016.
- [34] F. A. Spanhol, L. S. Oliveira, P. R. Cavalin, C. Petitjean, and L. Heutte, A. B. Banff, "Deep features for breast cancer histopathological image classification," in *Proceedings of the International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 1868-1873, Canada, CA, USA, 2017.
- [35] Z. Han, B. Wei, Y. Zheng, Y. Yin, K. Li, and S. Li, "Breast cancer multi-classification from histopathological images with structured deep learning model," *Scientific Reports*, vol. 7, no. 1, p. 4172, 2017.
- [36] R. Man, P. Yang, and B. Xu, "Classification of breast cancer histopathological images using discriminative patches screened by generative adversarial networks," *IEEE Access*, vol. 8, Article ID 155362, 2020.
- [37] A. Kumar, S. K. Singh, S. Saxena et al., "Deep feature learning for histopathological image classification of canine mammary tumors and human breast cancer," *Information Sciences*, vol. 508, pp. 405-421, 2020.
- [38] N. Bayramoglu, J. Kannala, and J. Heikkilä, "Deep learning for magnification independent breast cancer histopathology image classification," in *Proceedings of the International Conference on Pattern Recognition (ICPR)*, pp. 2440-2445, Cancun, Mexico, December 2016.
- [39] J. Sun and A. Binder, "Comparison of deep learning architectures for H&E histopathology images," in *Proceedings of the IEEE Conference on Big Data and Analytics (ICBDA)*, pp. 43-48, Kuching, Malaysia, November 2017.
- [40] C. Kaushal and A. Singla, "Automated segmentation technique with self-driven post-processing for histopathological breast cancer images," *CAAI Trans. Intell. Technol.* vol. 5, no. 4, pp. 294-300, 2020.
- [41] M. Talo, "Convolutional neural networks for multi-class histopathology image classification," 2019, <https://arxiv.org/ftp/arxiv/papers/1903/1903.10035.pdf>.
- [42] H. Elmannai, M. Hamdi, and A. AlGarni, "Deep learning models combining for breast cancer histopathology image classification," *International Journal of Computational Intelligence Systems*, vol. 14, no. 1, pp. 1003-1013, 2021.
- [43] Z. Hameed, S. Zahia, B. Garcia-Zapirain, J. Javier Aguirre, and A. María Vanegas, "Breast cancer histopathology image classification using an ensemble of deep learning models," *Sensors*, vol. 20, no. 16, p. 4373, 2020.
- [44] D. A. Ragab, M. Sharkas, S. Marshall, and J. Ren, "Breast cancer detection using deep convolutional neural networks and support vector machines," *PeerJ*, vol. 7, 2019.
- [45] F. P. Romero, A. Tang, and S. Kadoury, "Multi-level batch normalization in deep networks for invasive ductal carcinoma cell discrimination in histopathology images," in *Proceedings of the International Symposium on Biomedical Imaging (ISBI)*, pp. 1092-1095, Venice, Italy, July 2019.
- [46] H. L. Minh, M. M. Van, and T. V. Lang, "Deep feature fusion for breast cancer diagnosis on histopathology images," in *Proceedings of the International Conference on Knowledge and Systems Engineering (KSE)*, pp. 1-6, Da Nang, Vietnam, September 2019.
- [47] Y. Liu, K. Gadeballi, M. Norouzi et al., "Detecting cancer metastases on gigapixel pathology images," 2017, <https://arxiv.org/abs/1703.02442>.
- [48] M. A. Alantari, S. M. Han, and T. S. Kim, "Evaluation of deep learning detection and classification towards computer-aided diagnosis of breast lesions in digital X-ray mammograms," *Computer Methods and Programs in Biomedicine*, vol. 196, Article ID 105584, 2020.
- [49] H. Zhang, R. Wu, T. Yuan et al., "DE-Ada\*: a novel model for breast mass classification using cross-modal pathological semantic mining and organic integration of multi-feature fusions," *Information Sciences*, vol. 539, pp. 461-486, 2020.
- [50] N. Brancati, G. De Pietro, D. Riccio, and M. Frucci, "Gigapixel histopathological image analysis using attention-based neural networks," *IEEE Access*, vol. 9, Article ID 87552, 2021.
- [51] E. Mahraban Nejad, L. S. Affendey, R. B. Latip, I. B. Ishak, and R. Banaeeyan, "Transferred semantic scores for scalable retrieval of histopathological breast cancer images," *International Journal of Multimedia Information Retrieval*, vol. 7, no. 4, pp. 241-249, 2018.
- [52] V. K. Reshma, N. Arya, S. S. Ahmad et al., "Detection of breast cancer using histopathological image classification dataset with deep learning techniques," *BioMed Research International*, vol. 202213 pages, Article ID 8363850, 2022.
- [53] H. Huang, X. Feng, J. Jiang, P. Chen, and S. Zhou, "Mask RCNN algorithm for nuclei detection on breast cancer histopathological images," *International Journal of Imaging Systems and Technology*, vol. 32, no. 1, pp. 209-217, 2022.
- [54] G. Jayandhi, J. Leena Jasmine, and S. Mary Joans, "Mammogram learning system for breast cancer diagnosis using deep learning SVM," *Computer Systems Science and Engineering*, vol. 40, no. 2, pp. 491-503, 2022.
- [55] S. Sharma and S. Kumar, "The Xception model: a potential feature extractor in breast cancer histology images classification," *ICT Express*, vol. 8, no. 1, pp. 101-108, 2022.
- [56] H. Zerouaoui and A. Idri, "Deep hybrid architectures for binary classification of medical breast cancer images," *Biomedical Signal Processing and Control*, vol. 71, Article ID 103226, 2022.
- [57] W. Zhi, H. W. F. Yeung, Z. Chen, S. M. Zandavi, Z. Lu, and Y. Y. Chung, "Using transfer learning with convolutional neural networks to diagnose breast cancer from histopathological images," *International Conference on Neural Information Processing (ICONIP), China*, vol. 10637, pp. 669-676, 2017.
- [58] J. Chang, J. Yu, T. Han, H. Chang, and E. Park, "A method for classifying medical images using transfer learning: a pilot study on histopathology of breast cancer," in *Proceedings of the International Conference on E-Health Networking*,

- Applications and Services (Healthcom)*, pp. 1–4, Dalian, China, June 2017.
- [59] S. A. Hassan, M. S. Sayed, M. I. Abdalla, and M. A. Rashwan, “Breast cancer masses classification using deep convolutional neural networks and transfer learning,” *Multimedia Tools and Applications*, vol. 79, no. 41–42, pp. 30735–30768, 2020.
- [60] M. F. I. Soumik, A. Z. B. Aziz, and M. A. Hossain, “Improved transfer learning based deep learning model for breast cancer histopathological image classification,” in *Proceedings of the 2021 International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI)*, pp. 1–4, Rajshahi, Bangladesh, June 2021.
- [61] H. Mahmoud, A. H. Alharbi, and D. S. Khafga, “Breast cancer classification using deep convolution neural network with transfer learning,” *Intelligent Automation & Soft Computing*, vol. 29, no. 3, pp. 803–814, 2021.
- [62] C. Munien and S. Viriri, “Classification of hematoxylin and eosin-stained breast cancer histology microscopy images using transfer learning with EfficientNets,” *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–17, Article ID 5580914, 2021.
- [63] S. Boumaraf, X. Liu, Z. Zheng, X. Ma, and C. Ferkous, “A new transfer learning based approach to magnification dependent and independent classification of breast cancer in histopathological images,” *Biomedical Signal Processing and Control*, vol. 63, Article ID 102192, 2021.
- [64] A. Saber, M. Sakr, O. M. Abo-Seida, A. Keshk, and H. Chen, “A novel deep-learning model for automatic detection and classification of breast cancer using the transfer-learning technique,” *IEEE Access*, vol. 9, pp. 71194–71209, 2021.
- [65] S. Soltane, S. Alshreef, and S. MSerag Eldin, “Classification and diagnosis of lymphoma’s histopathological images using transfer learning,” *Computer Systems Science and Engineering*, vol. 40, no. 2, pp. 629–644, 2022.
- [66] M. Alruwaili and W. Gouda, “Automated breast cancer detection models based on transfer learning,” *Sensors*, vol. 22, no. 3, p. 876, 2022.
- [67] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2261–2269, Honolulu, HI, USA, July 2017.
- [68] S. J. Pan and Q. Yang, “A survey on transfer learning,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.
- [69] S. Z. Ramadan, “Methods used in computer-aided diagnosis for breast cancer detection using mammograms: a review,” *Journal of Healthcare Engineering*, vol. 2020, pp. 1–21, Article ID 9162464, 2020.
- [70] Y. Benhammou, B. Achchab, F. Herrera, and S. Tabik, “BreakHis based breast cancer automatic diagnosis using deep learning: taxonomy, survey and insights,” *Neuro-computing*, vol. 375, pp. 9–24, 2020.
- [71] D. A. Zebari, D. A. Ibrahim, D. Q. Zeebaree et al., “Systematic review of computing approaches for breast cancer detection based computer aided diagnosis using mammogram images,” *Applied Artificial Intelligence*, vol. 35, no. 15, pp. 2157–2203, 2021.
- [72] K. R. Weiss, T. M. Khoshgoftaar, and D. Wang, “A survey of transfer learning,” *J. Big Data*, vol. 3, no. 1, p. 9, 2016.
- [73] M. A. Mohammed, B. Al-Khateeb, A. N. Rashid, D. A. Ibrahim, M. K. Abd Ghani, and S. A. Mostafa, “Neural network and multi-fractal dimension features for breast cancer classification from ultrasound images,” *Computers & Electrical Engineering*, vol. 70, pp. 871–882, 2018.
- [74] WHO, “Breast Cancer,” 2021, <https://www.who.int/news-room/fact-sheets/detail/breast-cancer>.
- [75] K. He, X. Zhang, S. Ren, and J. Sun, “Identity Mappings in Deep Residual Networks,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, vol. 9908, pp. 630–645, Netherlands, Europe, 2016.
- [76] N. B. C. Foundation, *Biopsy*, The National Breast Cancer Foundation, Texas, TX, USA, 2018.
- [77] Q. Abbas, “DeepCAD: a computer-aided diagnosis system for mammographic masses using deep invariant features,” *Computers*, vol. 5, no. 4, p. 28, 2016.
- [78] G. Aresta, T. Araujo, S. Kwok et al., “BACH: grand challenge on breast cancer histology images,” *Medical Image Analysis*, vol. 56, pp. 122–139, 2019.
- [79] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [80] A. Chan and J. A. Tuszynski, “Automatic prediction of tumour malignancy in breast cancer with fractal dimension,” *Royal Society Open Science*, vol. 3, no. 12, Article ID 160558, 2016.
- [81] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems 25,” in *Proceedings of the 26th Annual Conference on Neural Information Processing Systems 2012*, pp. 1106–1114, Lake Tahoe, Nevada, USA, October 2012.
- [82] H. Bay, T. Tuytelaars, and L. V. Gool, “SURF: Speeded Up Robust Features,” in *Proceedings of the European Conference on Computer Vision ECCV*, vol. 3951, pp. 404–417, Graz, Austria, July 2006.
- [83] Z. Guo, L. Zhang, and D. Zhang, “Rotation invariant texture classification using LBP variance (LBPV) with global matching,” *Pattern Recognition*, vol. 43, no. 3, pp. 706–719, 2010.
- [84] S. F. University, “Digital Database for Screening Mammography,” 2006, <http://www.eng.usf.edu/cvprg/mammography/database.html>.
- [85] J. Suckling and J. Parker, “Mammographic Image Analysis Society (MIAS) Database v1.21 [Dataset],” 2015, <https://www.repository.cam.ac.uk/handle/1810/250394>.
- [86] C. Szegedy, W. Liu, Y. Jia et al., “Going deeper with convolutions,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, Boston, MA, USA, July 2015.
- [87] L. Shamir, N. Orlov, D. Mark Eckley, T. J. Macura, and I. G. Goldberg, “IICBU 2008: a proposed benchmark suite for biological image analysis,” *Medical, & Biological Engineering & Computing*, vol. 46, no. 9, pp. 943–947, 2008.
- [88] C. Szegedy, W. Liu, Y. Jia et al., “Going deeper with convolutions,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, Boston, MA, USA, July 2015.
- [89] M. A. Kahya, W. Al-Hayani, and Z. Y. Algamal, “Classification of breast cancer histopathology images based on adaptive sparse support vector machine,” *Journal of Applied Mathematics and Bioinformatics*, vol. 7, pp. 49–69, 2017.
- [90] V. Gupta and A. Bhavsar, “An integrated multi-scale model for breast cancer histopathological image classification with joint colour-texture features,” in *Proceedings of the International Conference on Computer Analysis of Images and Patterns (CAIP)*, vol. 10425, pp. 354–366, Ystad, Sweden, August 2017.

- [91] N. Dhungel, G. Carneiro, and A. P. Bradley, "A deep learning approach for the analysis of masses in mammograms with minimal user intervention," *Medical Image Analysis*, vol. 37, pp. 114–128, 2017.
- [92] I. C. Moreira, I. Amaral, I. Domingues, A. Cardoso, M. J. Cardoso, and J. S. Cardoso, "INbreast: toward a full-field digital mammographic database," *Academic Radiology*, vol. 19, no. 2, pp. 236–248, 2012.
- [93] Y. Jia, E. Shelhamer, J. Donahue et al., "Caffe: Convolutional Architecture for Fast Feature Embedding," in *Proceedings of the ACM Int. Conf. on Multimedia (MM)*, pp. 675–678, Florida, FL, USA, 2014.
- [94] S. Kaymak, A. Helwan, and D. Uzun, "Breast cancer image classification using artificial neural networks. Procedia Computer Science," in *Proceedings of the International Conference on Theory and Application of Soft Computing, Computing with Words and Perception (ICSCCW)*, vol. 120, pp. 126–131, Budapest, Hungary, June 2017.
- [95] R. Hecht-Nielsen, "Theory of the backpropagation neural network," *Neural Networks*, vol. 1, pp. 445–448, 1988.
- [96] C. Bishop, "Improving the generalization properties of radial basis function neural networks," *Neural Computation*, vol. 3, no. 4, pp. 579–588, 1991.
- [97] Camelyon16, "Challenge on Cancer Metastases Detection in Lymph Node," 2016, <https://camelyon16.grand-challenge.org>.
- [98] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, July 2015.
- [99] S. J. Stanford, "Tissue Microarray Database," 2021, <https://tma.im/cgi-bin/home.pl>.
- [100] A. A. Nahid, A. Mikaelian, and Y. Kong, "Histopathological breast-image classification with restricted Boltzmann machine along with backpropagation," *Biomedical Research*, vol. 29, pp. 2068–2077, 2018.
- [101] W. Lu, C. S. Leung, and J. Sum, "Analysis on noisy Boltzmann machines and noisy restricted Boltzmann machines," *IEEE Access*, vol. 9, pp. 112955–112965, 2021.
- [102] H. Tamura, S. Mori, and T. Yamawaki, "Textural features corresponding to visual perception," *IEEE Trans. Syst. Man Cybern.* vol. 8, no. 6, pp. 460–473, 1978.
- [103] J. A. Badejo, E. Adetiba, A. Akinrinmade, and M. B. Akanle, "Medical image classification with hand-designed or machine-designed texture descriptors: a performance evaluation," *International Work-Conference on Bioinformatics and Biomedical Engineering (IWBBIO), Granada, Spain*, vol. 10814, pp. 266–275, 2018.
- [104] P. Alirezazadeh, B. Hejrati, A. Monsef-Esfahani, and A. Fathi, "Representation learning-based unsupervised domain adaptation for classification of breast cancer histopathology images," *Biocybernetics and Biomedical Engineering*, vol. 38, no. 3, pp. 671–683, 2018.
- [105] J. Spencer and K. S. John, "Random sparse bit strings at the threshold of adjacency," in *Proceedings of the Annual symposium on theoretical aspects of computer science (STACS)*, vol. 1373, pp. 94–104, Paris, France, October 1998.
- [106] C. Y. Maa and M. A. Shanblatt, "Linear and quadratic programming neural network analysis," *IEEE Transactions on Neural Networks*, vol. 3, no. 4, pp. 580–594, 1992.
- [107] F. Giannakas, C. Troussas, A. Krouska, C. Sgouropoulou, and I. Voyiatzis, "XGBoost and deep neural network comparison: the case of teams' performance," *International Conference on Intelligent Tutoring Systems (ITS), Virtual Event*, vol. 12677, pp. 343–349, 2021.
- [108] D. S. Morillo, J. Gonzalez, M. G. Rojo, and J. Ortega, "Classification of breast cancer histopathological images using KAZE features," *Int. Work-Conference on Bioinformatics and Biomedical Engineering (IWBBIO), Spain*, vol. 10814, pp. 276–286, 2018.
- [109] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "KAZE Features," in *Proceedings of the European Conference on Computer Vision (ECCV)*, vol. 7577, pp. 214–227, Florence, Italy, June 2012.
- [110] S. Chatteraj and K. Vishwakarma, "Classification of histopathological breast cancer images using iterative VMD aided Zernike moments & textural signatures," 2018, <https://arxiv.org/abs/1801.04880>.
- [111] T. Theodoridis, K. Loumpoulias, N. Vretos, and P. Daras, "Zernike pooling: generalizing average pooling using zernike moments," *IEEE Access*, vol. 9, pp. 121128–121136, 2021.
- [112] N. Lassance and F. Vrans, "Minimum Rényi entropy portfolios," *Annals of Operations Research*, vol. 299, no. 1–2, pp. 23–46, 2021.
- [113] M. Rahimi and M. Mohammadi Anjedani, "A local view on the Hudetz correction of the Yager entropy of dynamical systems," *International Journal of General Systems*, vol. 48, no. 3, pp. 321–333, 2019.
- [114] Y. Zheng, Z. Jiang, H. Zhang et al., "Size-scalable content-based histopathological image retrieval from database that consists of WSIs," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 4, pp. 1278–1287, 2018.
- [115] R. W. Hamming, "Error detecting and error correcting codes," *Bell System Technical Journal*, vol. 29, no. 2, pp. 147–160, 1950.
- [116] R. Mukkamala, P. S. Neeraja, S. Pamidi, T. Babu, and T. Singh, "Deep PCANet framework for the binary categorization of breast histopathology images," in *Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 105–110, India, September 2018.
- [117] T. H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "PCANet: a simple deep learning baseline for image classification?" *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5017–5032, 2015.
- [118] A. Rakhlin, A. Shvets, V. Igloukov, and A. A. Kalinin, "Deep convolutional neural networks for breast cancer histology image analysis," in *Proceedings of the International Conference on Image Analysis and Recognition (ICIAR)*, vol. 10882, pp. 737–744, Portugal, July 2018.
- [119] M. A. Almasni, M. A. Alantari, J. M. Park et al., "Simultaneous detection and classification of breast masses in digital mammograms via a deep learning YOLO-based CAD system," *Computer Methods and Programs in Biomedicine*, vol. 157, pp. 85–94, 2018.
- [120] A. G. Howard, M. Zhu, B. Chen et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," 2017, <https://arxiv.org/abs/1704.04861>.
- [121] B. S. Veeling, J. Linmans, J. Winkens, T. Cohen, and M. Welling, "Rotation equivariant CNNs for digital pathology," *Int. Conf. on Medical Image Computing and Computer Assisted Intervention (MICCAI), Spain*, vol. 11071, pp. 210–218, 2018.
- [122] A. Pego and P. Aguiar, "Bioimaging Challenge 2015 Breast Histology Dataset," 2015, <http://www.bioimaging2015.ineb.up.pt/dataset.html>.

- [123] R. Lenz and P. L. Carmona, "Transform Coding of RGB-Histograms," in *Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP)*, pp. 117–124, Lisboa, Portugal, November 2009.
- [124] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation networks," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7132–7141, Salt Lake City, UT, USA, May 2018.
- [125] D. M. Vo, N. Q. Nguyen, and S. W. Lee, "Classification of breast cancer histology images using incremental boosting convolution networks," *Information Sciences*, vol. 482, pp. 123–138, 2019.
- [126] M. Babaie, S. Kalra, A. Sriram et al., "Classification and Retrieval of Digital Pathology Scans," in *Proceedings of the A New Dataset. Conference on Computer Vision and Pattern Recognition Workshops*, pp. 760–768, CVPR Workshops, Honolulu, HI, USA, July 2017.
- [127] X. Li, M. Radulovic, K. Kanjer, and K. N. Plataniotis, "Discriminative pattern mining for breast cancer histopathology image classification via fully convolutional autoencoder," *IEEE Access*, vol. 7, pp. 36433–36445, 2019.
- [128] R. S. Lee, F. Gimenez, A. Hoogi, K. K. Miyake, M. Gorovoy, and D. L. Rubin, "A curated mammography data set for use in computer-aided detection and diagnosis research," *Scientific Data*, vol. 4, no. 1, Article ID 170177, 2017.
- [129] C. Roa, "Data from: High-Throughput Adaptive Sampling for Whole-Slide Histopathology Image Analysis (HASHI) via Convolutional Neural Networks: Application to Invasive Breast Cancer Detection," 2018, <https://datadryad.org/stash/dataset/doi:10.5061/dryad.1g2nt41>.
- [130] P. Stanitsas, A. Cherian, V. Morellas, R. Tejpaul, N. Papanikolopoulos, and A. Truskinovsky, "Image descriptors for weakly annotated histopathological breast cancer data," *Frontiers in Digital Health*, vol. 2, Article ID 572671, 2020.
- [131] M. A. Asan and A. Ozsoy, "cuRCD: region covariance descriptor CUDA implementation," *Multimedia Tools and Applications*, vol. 80, no. 13, pp. 19737–19751, 2021.
- [132] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial Intelligence*, vol. 89, no. 1-2, pp. 31–71, 1997.
- [133] A. H. Fischer, K. A. Jacobson, J. Rose, and R. Zeller, "Hematoxylin and eosin staining of tissue and cell sections," *CSH Protocols*, vol. 2008, 2008.
- [134] S. K. Asare, F. You, and O. T. Nartey, "A semisupervised learning scheme with self-paced learning for classifying breast cancer histopathological images," *Computational Intelligence and Neuroscience*, vol. 2020, pp. 1–16, Article ID 8826568, 2020.
- [135] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1800–1807, Honolulu, HI, USA, May 2017.
- [136] K. Dimitropoulos, P. Barmpoutis, C. Zioga, A. Kamas, K. Patsiaoura, and N. Grammalidis, "Grading of invasive breast carcinoma through Grassmannian VLAD encoding," *PLoS One*, vol. 12, no. 9, Article ID e0185110, 2017.
- [137] A. Janowczyk and A. Madabhushi, "Grading of invasive breast carcinoma through Grassmannian VLAD encoding," *Journal of Pathology Informatics*, vol. 7, Article ID 27563488, 2016.
- [138] Y. Feng, L. Zhang, and J. Mo, "Deep manifold preserving autoencoder for classifying breast cancer histopathological images," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 17, no. 1, pp. 91–101, 2020.
- [139] E. J. Zaferani, M. Teshnehlab, and M. Vali, "Automatic personality traits perception using asymmetric auto-encoder," *IEEE Access*, vol. 9, pp. 68595–68608, 2021.
- [140] Y. LeCun, B. E. Boser, J. S. Denker et al., "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.
- [141] R. H. Carvalho, A. S. Martins, L. A. Neves, and M. Z. do Nascimento, "Analysis of features for breast cancer recognition in different magnifications of histopathological images," in *Proceedings of the International Conference on Systems, Signals and Image Processing (IWSSIP)*, pp. 39–44, Brazil, August 2020.
- [142] M. H. Sharif and C. Djeraba, "An entropy approach for abnormal activities detection in video streams," *Pattern Recognition*, vol. 45, no. 7, pp. 2543–2561, 2012.
- [143] A. Toomaj and H. A. Atabay, "Some new findings on the cumulative residual Tsallis entropy," *Journal of Computational and Applied Mathematics*, vol. 400, Article ID 113669, 2022.
- [144] J. Li, J. Zhang, Q. Sun et al., "Breast cancer histopathological image classification based on deep second-order pooling network," in *Proceedings of the Int. Joint Conf. On Neural Networks (IJCNN)*, pp. 1–7, London, U K, November 2020.
- [145] P. Li, J. Xie, Q. Wang, and Z. Gao, "Towards Faster Training of Global Covariance Pooling Networks by Iterative Matrix Square Root Normalization," in *Proceedings of the Conference on computer vision and pattern recognition (CVPR)*, pp. 947–955, Salt Lake City, UT, USA, September 2018.
- [146] L. Shen, L. R. Margolies, J. H. Rothstein, E. Fluder, R. McBride, and W. Sieh, "Deep learning to improve breast cancer detection on screening mammography," *Scientific Reports*, vol. 9, no. 1, Article ID 12495, 2019.
- [147] G. Li, C. Li, G. Wu, D. Ji, and H. Zhang, "Multi-view attention-guided multiple instance detection network for interpretable breast cancer histopathological image diagnosis," *IEEE Access*, vol. 9, pp. 79671–79684, 2021.
- [148] R. Yan, F. Ren, Z. Wang et al., "Breast cancer histopathological image classification using a hybrid deep neural network," *Methods*, vol. 173, pp. 52–60, 2020.
- [149] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules Advances in Neural Information Processing Systems 30," in *Proceedings of the Annual Conference on Neural Information Processing Systems*, pp. 3856–3866, Long Beach, CA, USA, April 2017.
- [150] J. Kundale and S. Dhage, "Classification of breast cancer using histology images: handcrafted and pre-trained features based approach," *IOP Conference Series: Materials Science and Engineering*, vol. 1074, no. 1, Article ID 012008, 2021.
- [151] A. R. H. Khayyat, X. Sun, and P. L. Rosin, "Improved DSIFT descriptor based copy-rotate-move forgery detection. Image and video technology - 7th pacific-rim symposium (PSIVT), auckland," *New Zealand*, vol. 9431, pp. 642–655, 2015.
- [152] J. Wang, J. Yang, K. Yu, F. Lv, T. S. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3360–3367, San Francisco, USA, January 2010.
- [153] O. Attallah, F. Anwar, N. M. Ghanem, and M. A. Ismail, "Histo-CADx: duo cascaded fusion stages for breast cancer diagnosis from histopathological images," *PeerJ Computer Science*, vol. 7, p. e493, 2021.

- [154] K. C. Burçak, Ö. K. Baykan, and H. Uguz, “A new deep convolutional neural network model for classifying breast cancer histopathological images and the hyperparameter optimisation of the proposed model,” *The Journal of Supercomputing*, vol. 77, no. 1, pp. 973–989, 2021.
- [155] H. Iimori, G. T. F. de Abreu, O. Taghizadeh, R. A. Stoica, T. Hara, and K. Ishibashi, “A stochastic gradient descent approach for hybrid mmWave beamforming with blockage and CSI-error robustness,” *IEEE Access*, vol. 9, pp. 74471–74487, 2021.
- [156] A. Botev, G. Lever, and D. Barber, “Nesterov’s accelerated gradient and momentum as approximations to regularised update descent,” in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pp. 1899–1903, Anchorage, AK, USA, August 2017.
- [157] A. Byerly and T. Kalganova, “Homogeneous vector capsules enable adaptive gradient descent in convolutional neural networks,” *IEEE Access*, vol. 9, pp. 48519–48530, 2021.
- [158] N. Shi, D. Li, M. Hong, and R. Sun, “RMSprop converges with proper hyper-parameter,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, Virtual Event, Austria, February 2021.
- [159] Z. Qu, S. Yuan, R. Chi, L. Chang, and L. Zhao, “Genetic optimization method of pantograph and catenary comprehensive monitor status prediction model based on adadelta deep neural network,” *IEEE Access*, vol. 7, pp. 23210–23221, 2019.
- [160] I. K. M. Jais, A. R. Ismail, and S. Q. Nisa, “Adam optimization algorithm for wide and deep neural network,” *Knowledge Engineering and Data Science*, vol. 2, no. 1, pp. 41–46, 2019.
- [161] I. Hirra, M. Ahmad, A. Hussain et al., “Breast cancer classification from histopathological images using patch-based deep learning modeling,” *IEEE Access*, vol. 9, pp. 24273–24287, 2021.
- [162] G. E. Hinton, S. Osindero, and Y. W. Teh, “A fast learning algorithm for deep belief nets,” *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.
- [163] Q. B. Baker and A. Abu Qutaish, “Evaluation of histopathological images segmentation techniques for breast cancer detection,” in *Proceedings of the International Conference on Information and Communication Systems (ICICS)*, pp. 134–139, Valencia, Spain, May 2021.
- [164] B. Ehteshami Bejnordi, M. Veta, P. Johannes van Diest et al., “Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer,” *JAMA*, vol. 318, no. 22, pp. 2199–2210, 2017.
- [165] M. Veta, Y. J. Heng, N. Stathonikos et al., “Predicting breast tumor proliferation from whole-slide images: the TUPAC16 challenge,” *Medical Image Analysis*, vol. 54, pp. 111–121, 2019.
- [166] M. Z. D. Nascimento, A. S. Martins, L. A. Neves, R. P. Ramos, E. L. Flôres, and G. A. Carrijo, “Classification of masses in mammographic image using wavelet domain features and polynomial classifier,” *Expert Systems with Applications*, vol. 40, no. 15, pp. 6213–6221, 2013.
- [167] M. Tan and Q. V. Le, “EfficientNet: rethinking model scaling for convolutional neural networks,” in *Proceedings of the 36th International Conference on Machine Learning (ICML)*, vol. 97, pp. 6105–6114, Long Beach, California, USA, December 2019.
- [168] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. F. Fei, “ImageNet: A Large-Scale Hierarchical Image Database,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255, Miami, Florida, USA, April 2009.
- [169] A. Ameh Joseph, M. Abdullahi, S. B. Junaidu, H. Hassan Ibrahim, and H. Chiroma, “Improved multi-classification of breast cancer histopathological images using handcrafted features and deep neural network (dense layer),” *Intelligent Systems with Applications*, vol. 14, Article ID 200066, 2022.
- [170] P. C. Chhipa, R. Upadhyay, G. G. Pihlgren, R. Saini, S. Uchida, and M. Liwicki, “Magnification prior: a self-supervised method for learning representations on breast cancer histopathological images,” 2022, <https://arxiv.org/abs/2203.07707>.
- [171] Y. Zou, J. Zhang, S. Huang, and B. Liu, “Breast cancer histopathological image classification using attention high-order deep network,” *International Journal of Imaging Systems and Technology*, vol. 32, no. 1, pp. 266–279, 2022.
- [172] M. Liu, Y. He, M. Wu, and C. Zeng, “Breast histopathological image classification method based on autoencoder and siamese framework,” *Information*, vol. 13, no. 3, p. 107, 2022.
- [173] D. A. Naik, R. M. Mohana, G. Ramu, Y. S. Lalitha, M. SureshKumar, and K. V. Raghavender, “Analyzing histopathological images by using machine learning techniques,” *Applied Nanoscience*, pp. 1–7, 2022.
- [174] S. Chattopadhyay, A. Dey, P. K. Singh, and R. Sarkar, “DRDA-Net: dense residual dual-shuffle attention network for breast cancer classification using histopathological images,” *Computers in Biology and Medicine*, vol. 145, Article ID 105437, 2022.
- [175] F. Shahidi, S. Mohd Daud, H. Abas, N. A. Ahmad, and N. Maarop, “Breast cancer classification using deep learning approaches and histopathology image: a comparison study,” *IEEE Access*, vol. 8, pp. 187531–187552, 2020.
- [176] C. Kampf, I. Olsson, U. Ryberg, E. Sjostedt, and F. Ponten, “Production of tissue microarrays, immunohistochemistry staining and digitalization within the human protein atlas,” *Journal of Visualized Experiments*, vol. 31, no. 63, p. 3620, 2012.
- [177] H. A. Shehu, M. H. Sharif, M. H. U. Sharif et al., “Deep sentiment analysis: a case study on stemmed Turkish twitter data,” *IEEE Access*, vol. 9, pp. 56836–56854, 2021.
- [178] M. H. Sharif, “An eigenvalue approach to detect flows and events in crowd videos,” *Journal of Circuits, Systems, and Computers*, vol. 26, no. 07, Article ID 1750110, 2017.
- [179] H. A. Shehu, M. H. Sharif, and R. A. Ramadan, “Distributed mutual exclusion algorithms for intersection traffic problems,” *IEEE Access*, vol. 8, pp. 138277–138296, 2020.
- [180] P. Jungkass and M. Berekovic, “Static allocation of basic blocks based on runtime and memory requirements in embedded real-time systems with hierarchical memory layout,” in *Proceedings of the Second Workshop on Next Generation Real-Time Embedded Systems*, vol. 87, no. 1–3, pp. 3–14, Budapest, Hungary, February 2021.
- [181] I. Loshchilov and F. Hutter, “SGDR: stochastic gradient descent with warm restarts,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, pp. 1–16, Toulon, France, September 2017.
- [182] C. Zhang, P. Benz, D. M. Argaw et al., “ResNet or DenseNet? Introducing Dense Shortcuts to ResNet,” in *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, pp. 3549–3558, HI, USA, December 2021.
- [183] I. L. S. V. R. C. ImageNet, “Large Scale Visual Recognition Challenge (ILSVRC),” 2010, <https://image-net.org/challenges/LSVRC>.



- [184] F. N. Iandola, M. W. Moskewicz, K. Ashraf, S. Han, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size," *CoRR*, 2016, <https://arxiv.org/abs/1602.07360?context=cs>.
- [185] S. Kornblith, J. Shlens, and Q. V. Le, "Do Better ImageNet Models Transfer Better?" in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2661–2671, Canada CA, USA, March 2019.
- [186] M. N. Gurcan, L. Boucheron, A. Can et al., "Histopathological image analysis: a review," *IEEE Rev. Biomed. Eng.*, vol. 2, pp. 147–171, 2009.
- [187] M. Macenko, M. Niethammer, J. S. Marron et al., "A method for normalizing histology slides for quantitative analysis," in *Proceedings of the International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 1107–1110, Boston, MA, USA, June 2009.
- [188] Keras, *Keras API*, 2021.
- [189] H. A. Shehu, R. A. Ramadan, and M. H. Sharif, "Artificial intelligence tools and their capabilities," *PLOMS AI*, p. 1, 2021.
- [190] D. P. Kingma, J. Ba, and Adam, "A Method for Stochastic Optimization," in *Proceedings of the International Conference on Learning Representations (ICLR)*, Y. Bengio and Y. LeCun, Eds., San Diego, CA, USA, May 2015.
- [191] N. I. H. Malaria, "Datasets of National Institutes of Health (NIH)," 2021, <https://www.kaggle.com/iarunava/cell-images-for-detecting-malaria>.
- [192] Kaggle, "CoronaHack - chest X-ray-dataset," 2021, <https://github.com/ieee8023/covid-chestxray-dataset>.
- [193] Kaggle and Skin Cancer, "Malignant vs. Benign," 2021, <https://www.kaggle.com/fanconic/skin-cancer-malignant-vs-benign>.
- [194] H. A. Shehu, W. Browne, and H. Eisenbarth, "An Adversarial Attacks Resistance-Based Approach to Emotion Recognition from Images Using Facial Landmarks," in *Proceedings of the IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, pp. 1307–1314, Naples, Italy, August 2020.
- [195] M. H. Sharif and C. Djeraba, "A simple method for eccentric event espial using mahalnobis metric. Progress in pattern recognition, image analysis, computer vision, and applications," in *Proceedings of the 14th iberoamerican conference on pattern recognition (CIARP)*, vol. 5856, pp. 417–424, Guadalajara, Mexico, April 2009.
- [196] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *Journal of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1937.
- [197] R. L. Iman and J. M. Davenport, "Approximations of the critical region of the fbietkan statistic," *Communications in Statistics - Theory and Methods*, vol. 9, no. 6, pp. 571–595, 1980.
- [198] J. L. Hodges and E. L. Lehmann, "Rank methods for combination of independent experiments in analysis of variance," *The Annals of Mathematical Statistics*, vol. 33, no. 2, pp. 482–497, 1962.
- [199] D. Quade, "Using weighted rankings in the analysis of complete blocks with additive block effects," *Journal of the American Statistical Association*, vol. 74, no. 367, pp. 680–683, 1979.
- [200] P. Westfall and S. Young, *Resampling-based Multiple Testing: Examples and Methods for P-Value Adjustment*, John Wiley & Sons, New Jersey, NY, USA, 2004.
- [201] O. J. Dunn, "Multiple comparisons among means," *Journal of the American Statistical Association*, vol. 56, no. 293, pp. 52–64, 1961.
- [202] S. Holm, "A simple sequentially rejective multiple test procedure," *Scandinavian Journal of Statistics*, vol. 6, pp. 65–70, 1979.
- [203] Y. Hochberg, "A sharper Bonferroni procedure for multiple tests of significance," *Biometrika*, vol. 75, no. 4, pp. 800–802, 1988.
- [204] G. Hommel, "A stagewise rejective multiple test procedure based on a modified Bonferroni test," *Biometrika*, vol. 75, no. 2, pp. 383–386, 1988.
- [205] G. Hommel and G. Bernhard, "A rapid algorithm and a computer program for multiple test procedures using logical structures of hypotheses," *Computer Methods and Programs in Biomedicine*, vol. 43, no. 3–4, pp. 213–216, 1994.
- [206] B. S. Holland and M. D. Copenhaver, "An improved sequentially rejective Bonferroni test procedure," *Biometrics*, vol. 43, no. 2, pp. 417–423, 1987.
- [207] D. M. Rom, "A sequentially rejective test procedure based on a modified Bonferroni inequality," *Biometrika*, vol. 77, no. 3, pp. 663–665, 1990.
- [208] H. Finner, "On a monotonicity problem in step-down multiple test procedures," *Journal of the American Statistical Association*, vol. 88, no. 423, pp. 920–923, 1993.
- [209] J. David Li, "A two-step rejection procedure for testing multiple hypotheses," *Journal of Statistical Planning and Inference*, vol. 138, no. 6, pp. 1521–1527, 2008.
- [210] P. Nemenyi, *Distribution-free Multiple Comparisons*, PhD thesis, Princeton University, New Jersey, NY, USA, 1963.
- [211] J. P. Shaffer, "Modified sequentially rejective multiple test procedures," *Journal of the American Statistical Association*, vol. 81, no. 395, pp. 826–831, 1986.
- [212] G. Bergmann and G. Hommel, "Improvements of general multiple test procedures for redundant systems of hypotheses," in *Multiple Hypotheses Testing*, P. Bauer and G. Hommel, Eds., pp. 100–115, Springer, New York, NY, USA, 1988.
- [213] S. García and F. Herrera, "An extension on "Statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons," *Journal of Machine Learning Research*, vol. 9, pp. 2677–2694, 2008.
- [214] G. University, "Soft Computing and Intelligent Information Systems," 2020, <https://sci2s.ugr.es/sicidm>.
- [215] B. Calvo and G. Santafé, "Scmamp: statistical comparison of multiple algorithms in multiple problems," *Rice Journal*, vol. 8, no. 1, pp. 248–256, 2016.
- [216] S. Cayir, G. Solmaz, H. Kusetogullari et al., "MITNET: a novel dataset and a two-stage deep learning approach for mitosis recognition in whole slide images of breast cancer tissue," *Neural Computing & Applications*, pp. 1–15, 2022.
- [217] Q. U. Ain, H. Al-Sahaf, B. Xue, and M. Zhang, "Generating knowledge-guided discriminative features using genetic programming for melanoma detection," *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 5, no. 4, pp. 554–569, 2021.

## Research Article

# Tuberculosis Detection in Chest Radiographs Using Spotted Hyena Algorithm Optimized Deep and Handcrafted Features

Seifedine Kadry <sup>1,2,3</sup>, Gautam Srivastava <sup>4,5</sup>, Venkatesan Rajinikanth <sup>6</sup>,  
Seungmin Rho <sup>7</sup>, and Yongsung Kim <sup>8</sup>

<sup>1</sup>Department of Applied Data Science, Noroff University College, Kristinasand 4612, Norway

<sup>2</sup>Department of Electrical and Computer Engineering, Lebanese American University, Byblos, Lebanon

<sup>3</sup>Artificial Intelligence Research Center (AIRC), College of Engineering and Information Technology, Ajman University, Ajman, UAE

<sup>4</sup>Department of Mathematics and Computer Science, Brandon University, Brandon, R7A 6A9, Canada

<sup>5</sup>Research Center for Interneural Computing, China Medical University, Taichung 40402, Taiwan

<sup>6</sup>Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai 602105, Tamil Nadu, India

<sup>7</sup>Department of Industrial Security, Chung-Ang University, Seoul, Republic of Korea

<sup>8</sup>Department of Technology Education, Chungnam National University, Daejeon, Republic of Korea

Correspondence should be addressed to Yongsung Kim; [kys1001@cuk.edu](mailto:kys1001@cuk.edu)

Received 17 February 2022; Revised 3 September 2022; Accepted 13 September 2022; Published 6 October 2022

Academic Editor: Andrea Loddo

Copyright © 2022 Seifedine Kadry et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Lung abnormality in humans is steadily increasing due to various causes, and early recognition and treatment are extensively suggested. Tuberculosis (TB) is one of the lung diseases, and due to its occurrence rate and harshness, the World Health Organization (WHO) lists TB among the top ten diseases which lead to death. The clinical level detection of TB is usually performed using bio-medical imaging methods, and a chest X-ray is a commonly adopted imaging modality. This work aims to develop an automated procedure to detect TB from X-ray images using VGG-UNet-supported joint segmentation and classification. The various phases of the proposed scheme involved; (i) image collection and resizing, (ii) deep-features mining, (iii) segmentation of lung section, (iv) local-binary-pattern (LBP) generation and feature extraction, (v) optimal feature selection using spotted hyena algorithm (SHA), (vi) serial feature concatenation, and (vii) classification and validation. This research considered 3000 test images (1500 healthy and 1500 TB class) for the assessment, and the proposed experiment is implemented using Matlab®. This work implements the pretrained models to detect TB in X-rays with improved accuracy, and this research helped achieve a classification accuracy of >99% with a fine-tree classifier.

## 1. Introduction

In the healthcare industry, there is a heavy diagnostic burden because of the steady increase in disease incidence in humans due to various reasons. The burden of disease detection can be reduced in hospitals by developing and implementing automated disease detection systems using artificial intelligence (AI) [1–5].

The lungs are one of the vital internal organs, and an infection in the lungs can cause severe illness, including death [6–8]. Tuberculosis (TB) is one of the severe lung

diseases caused by *Mycobacterium tuberculosis* (*M. tuberculosis*), and it can cause severe breathing problems in human patients. Therefore, it is imperative to detect and treat *tuberculosis* in a timely manner. It is also a communicable illness that will affect a human quickly and easily if one has a weak immune system.

A recent report by World Health Organization (WHO) lists TB as one of the top 10 causes of death globally and the foremost reason for death from a solitary infectious agent [9]. This report also confirms that, in 2019, TB caused 1.4 million deaths worldwide, and this report estimated that ten

million people would be diagnosed with TB. Most people infected with TB (> 90%) are adults, and the infection rate in males is higher than in women. Increased TB rate in a country is due to poverty, which causes financial distress, susceptibility, marginalization, and bias in TB-infected people. Furthermore, this report also verifies that about a quarter of the world's population is infected with TB. Usually, TB is curable and preventable when diagnosed in its early phase, and >85% who develop TB can be completely recovered with a 6-month drug regimen [10, 11].

The clinical level diagnosis of TB is usually performed with various clinical tests, including the bio-images. The infected lung section is typically recorded using computed tomography (CT) and radiographs (X-ray). The recorded image is then examined using a computer algorithm or by an experienced doctor to identify the harshness of TB. The former research on TB detection confirms that early diagnosis is essential to reduce the disease burden; hence, the researchers suggest several automated diagnostic procedures [12, 13]. In literature, the detection of TB with chest X-ray is widely discussed due to its clinical significance. Several machine learning (ML) and deep learning (DL) procedures are developed and employed to assess chest X-ray pictures. The DL-supported scheme helps to achieve a better detection accuracy compared to the ML, and hence, the DL-supported TB detection is considered in this research. The proposed research proposes a TB detection framework using the pretrained DL scheme, which implements combined segmentation and classification to achieve better detection, as discussed in [14]. The earlier work by Rahman et al. [14] implemented UNet for the segmentation and pretrained DL schemes for the classification. In the earlier work, the performance of VGG16 is not discussed, and hence, this research attempted the detection of TB using the VGG-UNet-based technique. The different stages of this framework consist of (i) image collection and resizing, (ii) implementation of pretrained VGG-UNet to segment the lung section from X-ray, (iii) collection of deep features (DF), (iv) local-binary-pattern (LBP) generation using different weights and LBP feature extraction, (v) spotted hyena algorithm (SHA) based DF and HF reduction, (vi) generating a new feature section with the serial concatenation of features, and (vii) binary classification and validation.

In this work, 3000 test images (1500 healthy and 1500 TB) are collected from the dataset provided by Rahman et al. [14, 15]. Initially, every test image is resized to  $224 \times 224 \times 3$  pixels, and the converted images are then evaluated using the pretrained VGG-UNet. UNet is a well-known convolutional neural network (CNN)-based encoder-decoder assembly, and the enhancement of this scheme is already reported in the literature. The enhancement methods, such as VGG-UNet [16] and ResNet [17], are already employed in which the encoder section is modified using the DL scheme. In the considered VGG-UNet, the well-known VGG16 architecture is considered to implement the encoder-decoder assembly, and the earlier work on this scheme can be accessed from [4]. In this work, the encoder section provides the necessary DF, and the decoder section supplies the segmented lung section, which is then considered to extract HF. The optimal value of DF and HF is

then identified using SHA, and then, a serial concatenation is considered to combine these optimal features (DF + HF). This feature vector is then considered to validate the performance of the binary classifier with a 5-fold cross-validation, and the employed scheme helped to achieve a classification accuracy of 98.73% with the fine-tree classifier.

The main contribution of this research includes the following:

- (i) Execution of CNN-based joint segmentation and classification is implemented using VGG16
- (ii) LBP pattern generation with various weights is presented
- (iii) SHA-based feature selection and serial feature concatenation is discussed

Other sections are arranged as follows: Section 2 shows earlier related work, Section 3 demonstrates methodology, and Sections 4 and 5 present the experimental outcome and conclusion of this research.

## 2. Related Research

Automated disease detection schemes are developed to reduce the diagnostic burden in hospitals, and most of these schemes also support the decision-making and treatment planning processes. In the literature, several ML and DL schemes are discussed to identify the TB from chest X-rays with the help of benchmark and clinically collected images. Every procedure aims to get better detection accuracy. This section summarizes chosen procedures employed to examine the X-ray, and the necessary information is presented in Table 1.

The research by Rahman et al. [14] employed a combined segmentation and thresholding concept to improve disease detection performance. This work employed the proposed technique on 7000 images (3500 healthy and 3500 TB class) and presented a detailed examination using various pretrained CNN methods in the literature. With an experimental investigation, the proposed work confirmed that joint segmentation and classification help to get a better disease diagnosis. With this motivation, the proposed work of this research also adopted the joint segmentation and classification concept to examine the TB from the database provided by Rahman et al. [15]. In the earlier work, the VGG16 was not employed for the segmentation and classification task. Hence, the proposed research work adopted the VGG-UNet scheme for the investigation, in which the VGG16 acts as the encoder unit. The experimental outcome of this study confirms that the proposed scheme worked well on the chest X-ray database and helped to achieve a classification accuracy of >99% with the fine-tree classifier.

## 3. Methodology

This research division shows the scheme developed to identify the TB by joint segmentation and classification task. First, the necessary test pictures are collected from a benchmark image database represented by Rahman et al. [15], and after the collection, every image is resized to a dimension of  $224 \times 224 \times 3$  pixels. After the resizing task,

TABLE 1: Summary of automated TB detection schemes employed to examine X-ray images.

Reference	Developed procedure
Rajaraman and Antani [18]	A customized DL system is proposed to examine the Shenzhen CXR pictures, and the proposed system provided an accuracy of 83.7%. However, this work confirms that implementing a customized DL approach is complex and time-consuming
Hwa et al. [19]	Examination of TB from X-ray using ensemble DL system and canny-edge detection is implemented and achieved better values of accuracy (89.77%), sensitivity (90.91%), and specificity (88.64%). However, the implementation of canny-edge detection along with the ensemble DL scheme needs a larger image preprocessing task, and it will increase the detection time
Wong et al. [20]	The development of a customized DL technique called TB-Net is proposed, and this work helped to achieve better performance measures, such as accuracy (99.86%), sensitivity (100%), and specificity (99.71%). This research also proposes a customary model, which is relatively more complex than the pretrained models
Hooda et al. [21]	Seven convolutional layers and three fully connected layer-based customized DL method are proposed for TB detection and achieved a classification accuracy of 94.73%
Rohilla et al. [22]	This work employed the conventional AlexNet and VGG16 methods to examine the X-ray images and attained an accuracy of >81%
Nguyen et al. [23]	X-ray diagnosis performance of pretrained DL schemes is presented, and the employed technique helped to provide better TB recognition
Afzali et al. [24]	The contour-based silhouette descriptor technique is employed to detect TB, and the selected features provided an accuracy of 92.86%
Stirenko et al. [25]	The CNN-based disease diagnosis with lossless and lossy data expansion is employed, and the proposed method offers a better TB diagnosis with X-ray pictures
Rahman et al. [14]	Implementation of combined CNN segmentation and categorization is presented to identify TB from X-ray images. This work implemented the classification task with and without segmentation and achieved a TB detection accuracy of 96.47% and 98.6%, respectively. This work also presented a detailed evaluation methodology for TB detection using various pretrained DL methods

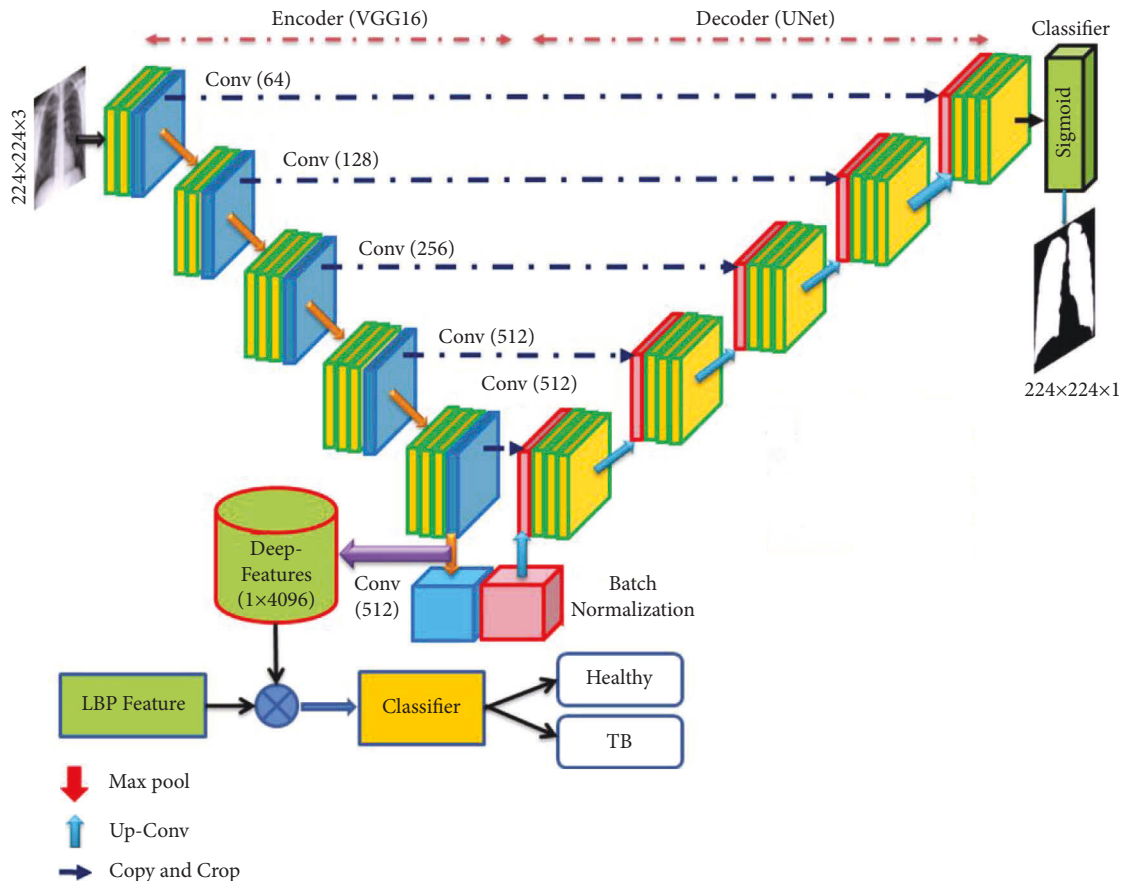


FIGURE 1: Joint segmentation and classification implemented for TB detection using X-ray.

TABLE 2: Chest X-ray image dataset information.

Class	Dimension	Total	Images	
			Training	Validation
Healthy	$224 \times 224 \times 3$	1500	1050	450
TB	$224 \times 224 \times 3$	1500	1050	450

every picture is evaluated by the VGG-UNet. Then, the encoder section presents the necessary DF, and the final layer (SoftMax) of the decoder section provides the binary form of the segmented lung section. The outcome of the encoder unit provides a DF of value, which is then reduced by  $1 \times 1 \times 1024$  using a chosen dropout rate (2 dropout layers with 50% dropout value to reduce  $1 \times 1 \times 4096$  to  $1 \times 1 \times 1024$ ), and these features are further reduced using the SHA to get the DF of a chosen dimension. The binary image obtained at the decoder section is then combined with its original test image to extract the lung section. The necessary LBP features are extracted from the extracted lung section, and these features are further reduced with SHA. Finally, a serial concatenation is then implemented to get DF + HF, and these features are then chosen to test and validate the performance of the developed system on the considered database.

The performance of the proposed scheme is tested using (i) DF alone and (ii) SHA-optimized DF + HF. During this assessment, the SoftMax-based binary classification is employed, and later, other binary classifiers existing in the literature are considered for testing the performance of the proposed scheme. The various stages presented in this scheme are shown in Figure 1. The concatenated feature is employed in this work to classify the X-ray images into healthy/TB classes.

**3.1. Image Dataset.** The merit of the automated disease diagnosis is then tested and verified using the clinically grade or benchmark medical data. In this research, the chest X-ray images considered by Rahman et al. [15] are adopted. From this dataset, 3000 images are collected to assess which 1500 images belong to the healthy group and the remaining 1500 with TB traces. Every collected image is resized to  $224 \times 224 \times 3$  pixels (approved size for VGG16). Of the total images, 70% (1050 images) are considered to train the developed scheme, and the remaining 30% (450 images) are considered for validation. The information about the test images is shown in Table 2, and the sample test images for the healthy/TB class are presented in Figure 2.

**3.2. Pretrained VGG-UNet.** Deep-learning-supported medical data assessment is a commonly employed technique, and most of these approaches are adopted to implement automatic segmentation and classification operations [26–28]. The CNN-based segmentation using the traditional UNet [29] and SegNet [30] is employed in the literature to extract and evaluate the disease-infected section from various modality medical images. The limitation of traditional CNN segmentation schemes is rectified by enhancing its performance using the pretrained DL schemes. The DL schemes

are considered to form the encoder and decoder section, which supports the feature extraction and segmentation for medical images of a chosen dimension. In this work, the pretrained VGG16 is then considered to implement the VGG-UNet scheme, and the necessary information about this architecture can be found in [4, 31].

Initially, the considered VGG-UNet is trained using X-ray images with the following tasks.

- (i) Predictable augmentation (rotation and zoom) to increase the number of training images
- (ii) Assignment of learning rate as  $1e-4$  for better accuracy
- (iii) Training with linear dropout rate (LDR) and Adam optimization

During this task, other vital parameters are assigned as follows: total iteration = 2000, total epochs = 50, dropout rates in the fully connected layer = 50%, and the final layer is the SoftMax unit with 5-fold cross-validation.

**3.3. Feature Extraction.** This section presents the outline of the DF and HF extraction procedure.

**3.3.1. Deep Feature.** The necessary deep features from the proposed scheme are extracted from the encoder section (VGG16) of the VGG-UNet. This section offered a feature vector of dimension  $1 \times 1 \times 4096$ , and it is then passed through three fully connected (FC) layers with a dropout rate of 50% to get a reduced feature vector of dimension  $1 \times 1 \times 1024$ . This feature is the DF, which is then considered to classify the X-ray images using a chosen binary classifiers. In this work, the classification task is executed using the conventional DF and the DF optimized using the SHA. The experimental outcome of this study confirms that the proposed work helped to get better classification accuracy with optimized DF compared to the conventional DF.

**3.3.2. Handcrafted Feature.** The HF is considered in ML-based automatic disease detection systems, and in this work, the HF is obtained using LBP of various weights as discussed in [32]. The various procedures to extract the HF from the chosen X-ray are as follows: the implemented VGG-UNet helps to extract the lung section in binary form. This binary image is then combined with its original test image to get the necessary lung section without the artifacts. After getting the lung image, the necessary LBP pattern is generated by assigning its weights as  $W = 1, 2, 3$  and  $d = 4$ , and from these images, the necessary LBP features with dimension  $1 \times 1 \times 59$  are extracted, and the extracted features are then optimized using the SHA.

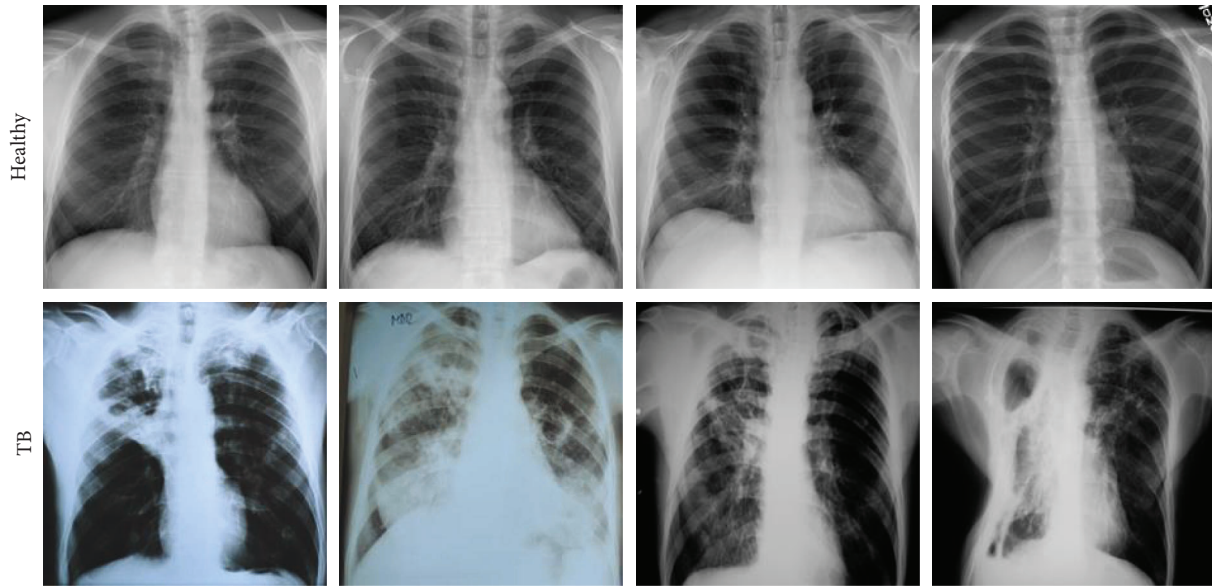


FIGURE 2: Sample X-ray images of healthy/TB class.

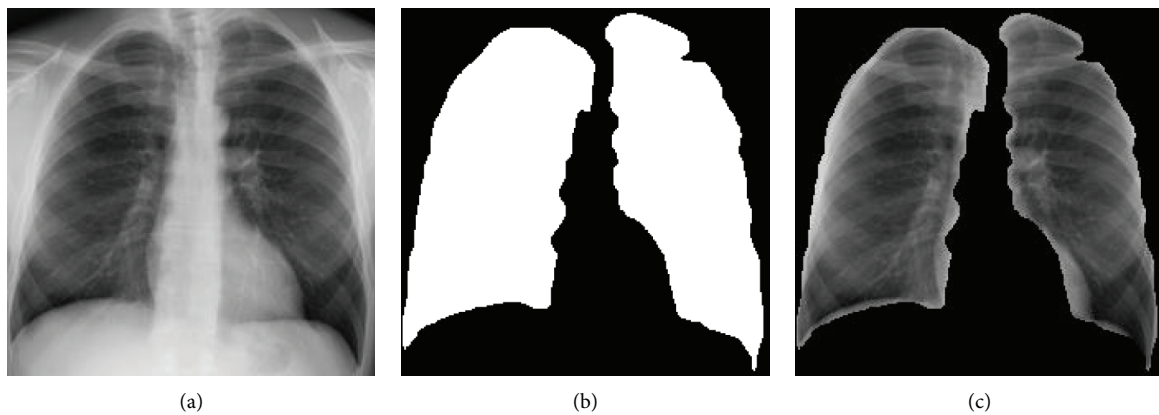


FIGURE 3: The result achieved with VGG-Unet and the extracted lung section. (a) Test image, (b) binary image, (c) section to be examined.

Figure 3 depicts the procedure to remove the artifact. Figures 3(a) and 3(b) show the sample test image and the extracted binary region by the decoder section of VGG-UNet. When Figures 3(a) and 3(b) are combined (pixel-wise multiplication), then we get Figure 3(c), which is the lung section without the artifact. This section is then considered to get the LBP pattern with various weights. The generated LBP pattern of a sample image is presented in Figure 4, in which Figures 4(a)–4(d) depict the outcome for the chosen weights. Every image provides 59 number of one-dimensional (1D) features, and the total features obtained with LBP are 236 features ( $59 \times 4$ ) which are then reduced using the SHA to avoid the overfitting difficulty in X-ray classification. Other information related to this task can be found in the earlier research works [4, 16].

**3.4. Feature Reduction with Spotted Hyena Algorithm.** Metaheuristic algorithms (MA) are adopted in the literature to find the finest solution for various real-world problems.

The earlier works related to medical image assessment confirm that the MA is widely adopted in various image examination works, such as thresholding, segmentation, and feature selection [33, 34]. The MA-based feature selection procedure is already discussed in various ML and DL techniques, and this procedure helps to get the finest feature vector, which avoided the overfitting problem during the automated classification. The MA-based feature selection can be used as an alternative technique for the traditional feature reduction procedures discussed in [35].

In this work, the feature reduction task is implemented for both the DF and HF using the SHA. It is a nature-motivated procedure invented in 2017 by mimicking the hunting events found in spotted hyena (SH) packs. The SH are the skillful animal that hunts as a pack, and this operation consists of the following stages: (i) choice making and following the prey, (ii) chasing the prey, (iii) surrounding the prey, and (iv) killing. The arithmetical replica developed by Dhiman and Kumar [36, 37] considered all constraints to improve the

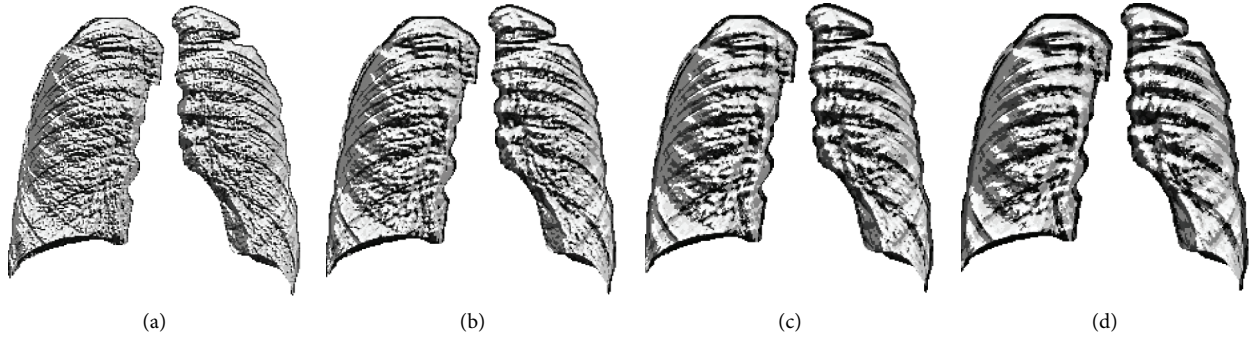


FIGURE 4: Generated LBP pattern of X-ray for various weights. (a)  $W = 1$ , (b)  $W = 2$ , (c)  $W = 3$ , (d)  $W = 4$ .

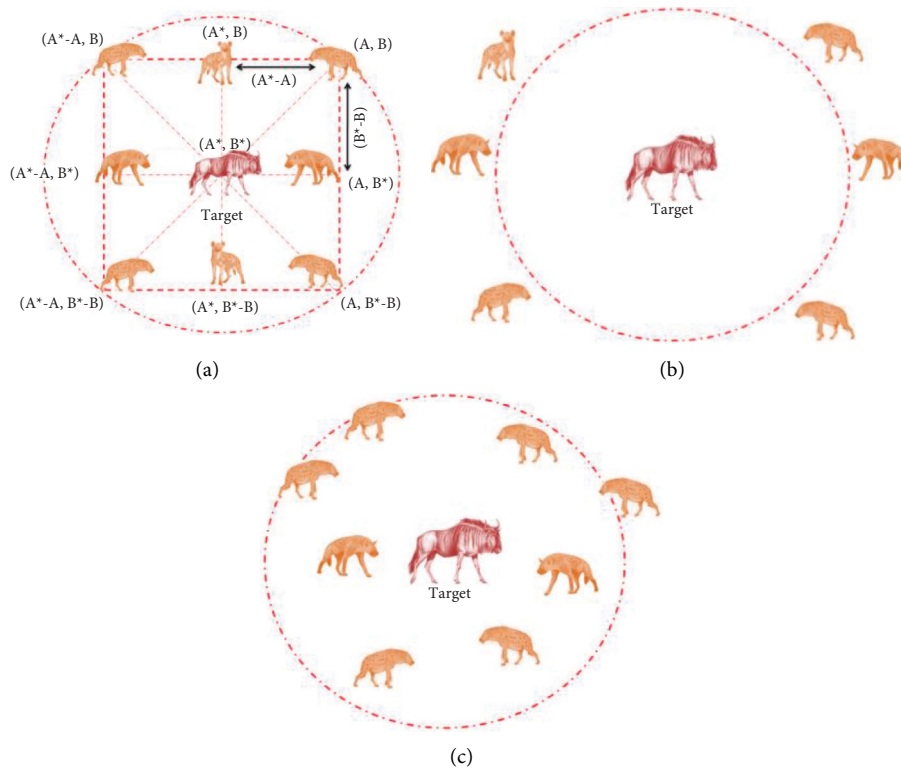


FIGURE 5: Various stages of SHA. (a) Encircling, (b) hunting, (c) attacking.

converge capability of the SHA. A similar kind of algorithm, known as the Dingo optimizer, is also developed and implemented by Bairwa et al. [38].

The various stages of the SHA are depicted in Figure 5, in which Figures 5(a)–5(c) present the operations, such as identifying and tracking the prey as in Figure 5(a), tracking and encircling the prey depicted in Figure 5(b), and hunting as presented in Figure 5(c). This operation is as follows: the leader in pack identifies the prey, and the leader and its pack will chase it till it is tired. When the prey is tired, the leader and its group will encircle the prey as depicted in Figure 5. In this context, every group member will adjust their location concerning the prey. This process is depicted in the figure

using notation A and B. This adjustment is carried out in the algorithm using mathematical operations such as multiplication and subtraction.

The encircling process is mathematically represented as follows:

$$\vec{D}_h = |\vec{B} \cdot \vec{P}_p(x) - \vec{P}(x)|, \quad (1)$$

$$\vec{P}(x+1) = \vec{P}_p(x) - \vec{E} \cdot \vec{D}_h, \quad (2)$$

where  $\vec{D}_h$  = distance among the hyena and prey,  $x$  = current iteration,  $\vec{P}_p$  = position vector of prey, and  $\vec{P}$  = position vector of hyena.

The coefficient vectors  $\vec{B}$  and  $\vec{E}$  are computed as follows:

$$\vec{B} = 2 \cdot \mathfrak{R} \vec{d}_1, \quad (3)$$

$$\vec{E} = 2 \vec{h} \cdot \mathfrak{R} \vec{d}_2 - \vec{h}, \quad (4)$$

$$\vec{h} = 5 - \left( \text{Iteration} * \left( \frac{5}{\text{Iter}_{\max}} \right) \right), \quad (5)$$

where  $\text{Iter}_{\max}$  = maximum iterations assigned,  $\vec{h}$  = a linearly decreasing value from 5 to 0 insteps of 0.1,  $\mathfrak{R} \vec{d}_1$  and  $\mathfrak{R} \vec{d}_2$  = random number [0, 1] number

In this figure, (A,B) are the hyena, and it will adjust its location towards the prey (A \*, B \*) based on the values of Eqns. (3) to (5).

In the hunting stage, the hyena pack will move close to the prey and proceed for the attack. This phase is represented as follows:

$$\vec{D}_h = |\vec{B} \cdot \vec{P}_h - \vec{P}_k|, \quad (6)$$

$$\vec{P}_k = \vec{P}_h - \vec{E} \cdot \vec{D}_h, \quad (7)$$

$$\vec{C}_h = \vec{P}_k + \vec{P}_{k+1} + \dots + \vec{P}_{k+N}, \quad (8)$$

where  $\vec{P}_h$  = leader which moves closer to prey and  $\vec{P}_k$  = positions of other hyenas in the pack, and  $N$  = total hyenas in the pack.

In the attacking phase, the hyena moves and attacks the prey, other hyenas in the group also follow the same technique, and the group attack will kill the prey. When the prey is dead, every hyena in the pack is on or nearer to the prey. This process is the convergence of the chosen agents towards the optimal location as in (7).

$$\vec{P}(x+1) = \frac{\vec{C}_h}{N}, \quad (9)$$

where  $\vec{P}(x+1)$  is the best position, in which every hyena of the pack converges. In this work, the SHA is initiated with the following parameters: number of hyena (agent) =  $N = 20$ , search dimension (D) = 2,  $\text{Iter}_{\max} = 2000$  and stopping criteria = maximization of Cartesian distance (CD) between features or  $\text{Iter}_{\max}$ .

The feature reduction with SHA is graphically depicted in Figure 6, and from this procedure, it is clear that this task selects the image features of the healthy/TB class based on the maximal CD, and this procedure is already discussed in earlier works [5, 34, 39]. As depicted in Figure 6, the SHA-based feature selection is separately employed to identify the optimal features, and after getting these features, it is then serially concatenated, and the concatenated features are then considered for classifier training and validation.

The number of DF and HF available for the optimization is depicted in the following equations:

$$DF_{VGG(1 \times 1 \times 1024)} = VGG_{(1,1)}, VGG_{(1,2)}, \dots, VGG_{(1,1024)}, \quad (10)$$

$$LBP1_{w1(1 \times 1 \times 59)} = W1_{(1,1)}, W1_{(1,2)}, \dots, W1_{(1,59)}. \quad (11)$$

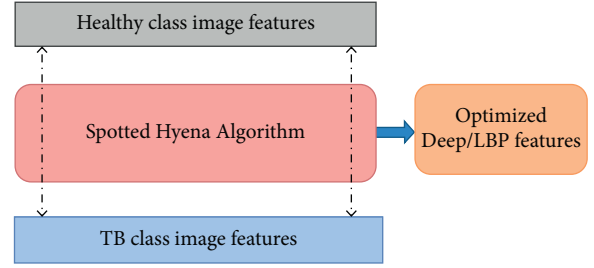


FIGURE 6: The SHA-based feature selection process.

$$LBP2_{w2(1 \times 1 \times 59)} = W2_{(1,1)}, W2_{(1,2)}, \dots, W2_{(1,59)}, \quad (12)$$

$$LBP3_{w3(1 \times 1 \times 59)} = W3_{(1,1)}, W3_{(1,2)}, \dots, W3_{(1,59)}, \quad (13)$$

$$LBP4_{w4(1 \times 1 \times 59)} = W4_{(1,1)}, W4_{(1,2)}, \dots, W4_{(1,59)}, \quad (14)$$

$$HF_{(1 \times 1 \times 236)} = LBP1 + LBP2 + LBP3 + LBP4. \quad (15)$$

During the feature reduction process, the DF in Eqn. (10) and HF in (8) are examined by the SHA, and the reduced features are then serially concatenated to get a hybrid feature vector (DF + HF). In this work, the SHA helps to reduce the DF to  $1 \times 1 \times 427$  and HF to  $1 \times 1 \times 166$ , and these features are then combined as in Eqn. (16) to get the new feature vector.

$$DF + HF_{(1 \times 1 \times 593)} = DF_{(1 \times 1 \times 427)} + HF_{(1 \times 1 \times 166)}. \quad (16)$$

The feature presented in (9) is then considered to train and test the classifiers considered in this study. The various binary classifiers considered in this research include Soft-Max, Naïve-Bayes (NB), random forest (RF), decision tree (DT) variants, K-nearest neighbors (KNN) variants, and SVM with linear kernel [40–43].

**3.5. Performance Validation.** The merit of an automated disease detection system is to be verified by computing the necessary performance values. In this work, the measures obtained from the confusion matrix are considered to confirm the eminence of the proposed scheme. These measures include true positive (TP), false negative (FN), true negative (TN), false positive (FP), accuracy (ACC), precision (PRE), sensitivity (SEN), specificity (SPE), and negative predictive value (NPV). The mathematical expressions of these values are presented in the following equations [42–46]:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}, \quad (17)$$

$$PRE = \frac{TP}{TP + FP}, \quad (18)$$

$$SEN = \frac{TP}{TP + FN}, \quad (19)$$

$$SPE = \frac{TN}{TN + FP}, \quad (20)$$



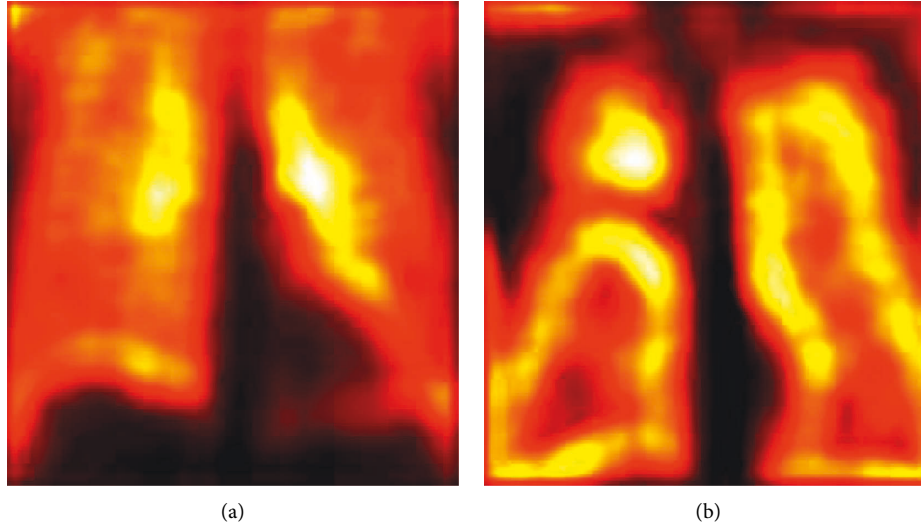


FIGURE 7: Sample test images obtained the convolution operation. (a) Normal, (b) TB.

TABLE 3: Classification result achieved with DF alone for a 5-fold cross-validation.

Classifier	TP	FN	TN	FP	ACC (%)	PRE (%)	SEN (%)	SPE (%)	NPV (%)
Fold 1	418	32	414	36	92.44	92.07	92.89	92.00	92.82
Fold 2	421	29	419	31	93.33	93.14	93.56	93.11	93.53
Fold 3	420	30	424	26	93.78	94.17	93.33	94.22	93.39
Fold 4	423	27	425	25	94.22	94.42	94.00	94.44	94.03
Fold 5	421	29	423	27	93.78	93.97	93.56	94.00	93.58

		SoftMax					SoftMax			
Output Class	Healthy	423 47.0%	25 2.8%	94.4% 5.6%	Output Class	Healthy	441 49.0%	8 0.9%	98.2% 1.8%	
	TB	27 3.0%	425 47.2%	94.0% 6.0%		TB	9 1.0%	442 49.1%	98.0% 2.0%	
			94.0% 6.0%	94.4% 5.6%		94.2% 5.8%			98.0% 2.0%	98.2% 1.8%
		Healthy	TB				Healthy	TB		
		Target Class					Target Class			
		(a)						(b)		

FIGURE 8: Confusion matrix attained with traditional and optimized features. (a) DF with SoftMax, (b) DF + HF with SoftMax.

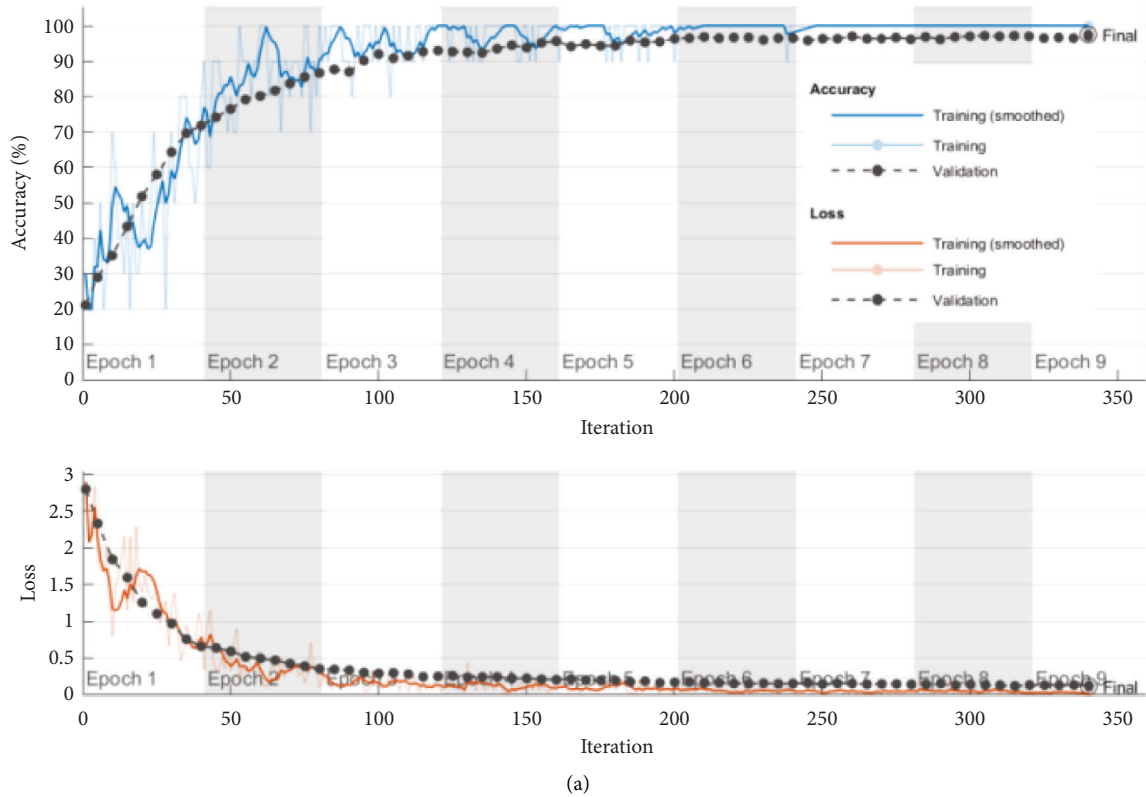
$$NPV = \frac{TN}{TN + FN}. \quad (21)$$

#### 4. Results and Discussion

This part of the research presents the present research's investigational outcome. This work is executed using a

workstation; Intel i7 2.9 GHz processor with 20 GB RAMS and 4 GB VRAM equipped with Matlab®.

Initially, the pretrained VGG-UNet scheme is trained using resized chest X-ray images till it extracts the lung section with better accuracy. After the training, its segmentation performance is tested using test images, and the outcome is recorded. Then, the extracted section is combined with the original image to extract the lung section



Fine-Tree

	Healthy	TB	
Healthy	446 49.6%	3 0.3%	99.3% 0.7%
TB	4 0.4%	447 49.7%	99.1% 0.9%
	99.1% 0.9%	99.3% 0.7%	99.2% 0.8%
	Healthy	TB	
	Target class		

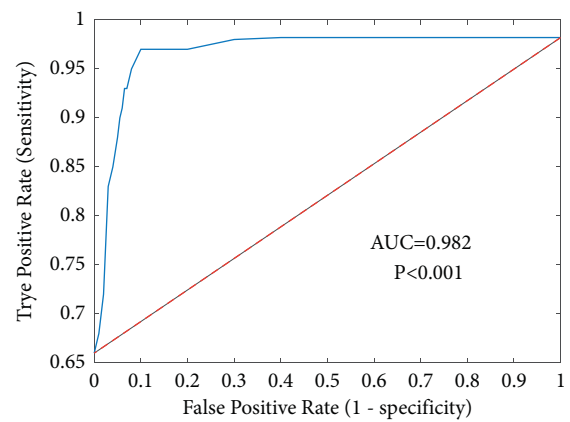


FIGURE 9: Investigational outcome of the fine-tree classifier with DF + HF. (a) Convergence, (b) confusion matrix, (c) ROC.

without the artifact, and the necessary LBP is generated when the HF is extracted. Similarly, the necessary DF is extracted from the encoder section (VGG16), and these features are then passed through the fully connected layers to get a feature vector of size  $1 \times 1 \times 1024$ . This procedure is similar to employing a traditional VGG16 scheme, and this feature is initially considered to classify the images with a SoftMax classifier, and the necessary performance is then recorded.

During the convolutional operation, the layers of the VGG-UNet help to recognize the necessary image features to support the necessary feature extraction and segmentation. The sample test image textures identified during a convolutional operation are presented in Figure 7. Figures 7(a)

and 7(b) depict the hot color map image obtained for healthy and TB class sample images, respectively. After extracting the necessary deep features from the test images (with a VGG16-like scheme), the necessary classification task is implemented using the SoftMax classifier with a 5-fold cross-validation. The achieved results are presented in Table 3. This table confirms that when the DF vector of dimension  $1 \times 1 \times 1024$  is considered, SoftMax provided a classification accuracy of 94.22%. This procedure is then repeated using the SHA-selected DF + HF presented in (9), and the achieved confusion matrix (CM) is presented in Figure 8. Figure 8(a) shows the CM of the DF case, and Figure 8(b) shows the CM of optimized DF + HF, and this confirms that the accuracy achieved with the proposed

TABLE 4: Sample results attained with the implemented UNet scheme.

Classifier	TP	FN	TN	FP	ACC (%)	PRE (%)	SEN (%)	SPE (%)	NPV (%)
SoftMax	441	9	442	8	98.11	98.22	98.00	98.22	98.00
NB	442	8	443	7	98.33	98.44	98.22	98.44	98.23
RF	441	9	443	7	98.22	98.44	98.00	98.44	98.01
Coarse tree	442	8	442	8	98.22	98.22	98.22	98.22	98.22
Medium tree	442	8	444	6	98.44	98.66	98.22	98.67	98.23
Fine tree	446	4	447	3	99.22	99.33	99.11	99.33	99.11
Coarse KNN	445	5	446	4	99.00	99.11	98.89	99.11	98.89
Medium KNN	444	6	445	5	98.78	98.88	98.67	98.89	98.66
Fine KNN	447	3	442	8	98.78	98.24	99.33	98.22	99.32
SVM linear	441	9	445	5	98.44	98.88	98.00	98.89	98.02

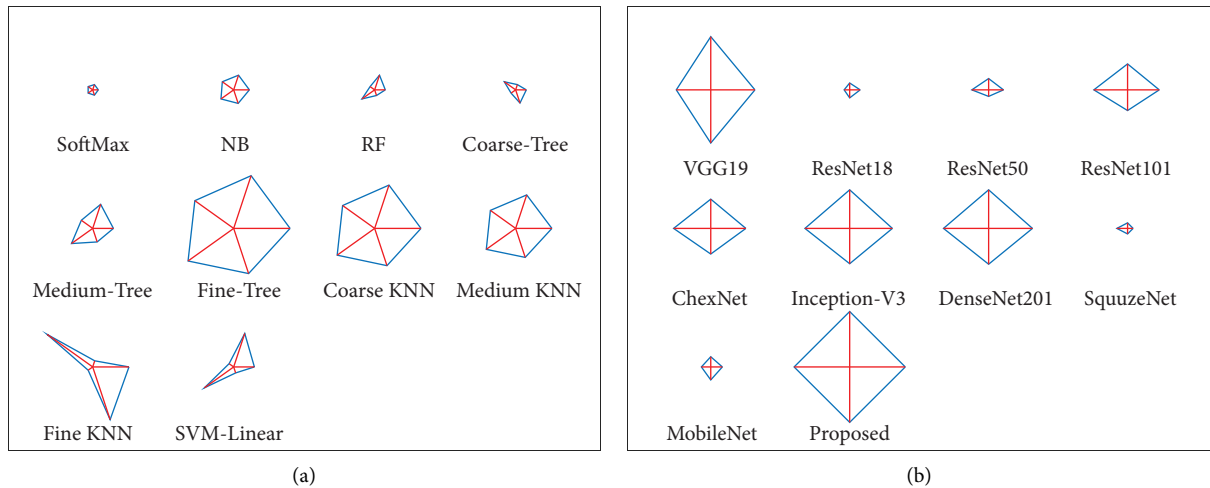


FIGURE 10: Overall performance of TB detection system demonstrated with glyph plot. (a) Results with DF + HF, (b) comparison with earlier work.

method is superior to the traditional technique. Hence, the performance of the proposed scheme is then confirmed with various binary classifiers using the DF + HF.

Figure 9 presents the experimental outcome achieved with the RF variant and the fine-tree classifier. Figure 9(a) depicts the convergence of the search, and Figures 9(b) and 9(c) show the CM and the receiver operating characteristic (ROC) curve, respectively. The results achieved with other chosen classifiers are presented in Table 4. This confirms that the classification accuracy of the fine tree is >99%, which confirms its merit over other techniques. In order to verify the performance of the proposed scheme, its best result is compared with the results of Rahman et al. [14] and confirmed that the proposed joint segmentation and classification scheme with SHA-selected DF + HF help to achieve a better outcome compared to the earlier works.

Table 4 confirms that the result of the fine tree is better than other binary classifiers, and the coarse KNN also helped to achieve a classification accuracy of 90% compared to other techniques. This confirmed that the optimized DF + HF supported classification helps to get a better overall result, as presented in Figure 10. Figure 10(a) shows the glyph plot for the overall performance of binary classifiers, and the pattern covering a maximum area is considered superior. The

comparison of the fine-tree classifier with earlier works is presented in Figure 10(b), which confirms its superiority over other classifiers. Figure 10(b) compares the ACC, PRE, SEN, and SPE of the earlier research by Rahman et al. [14] with the fine-tree result, and this comparison confirms that the proposed system's outcome is better.

This research implemented a joint segmentation and classification scheme to detect TB from chest X-rays with better accuracy. The main limitation of the proposed scheme is that it considered the artifact-removed image for getting the necessary HF from the LBP images. In the future, the LBP can be combined with other HF existing in the literature. Furthermore, the performance of the proposed scheme can be tested and verified with other benchmark chest X-ray images with various lung abnormalities.

## 5. Conclusion

In humans, TB is a severe disease that widely affects the lungs, and early diagnosis and treatment will help to reduce the severity. Furthermore, the timely detection and recommended medication will help to cure the TB completely. Due to its significance, a considerable number of research works are performed by researchers to support the

automated diagnosis. This research aims to develop and implement a joint segmentation and classification scheme with the help of a pretrained VGG-UNet scheme. The VGG-UNet system consists of an encoder-decoder assembly, in which the encoder helps to get the necessary DL features as in the traditional VGG16 system, and the decoder associated with the SoftMax classifier helps to extract the binary form of the lung image. This work considered the LBP pattern of the lung image to extract the necessary HF. This work considered the LBP with varied weights and helped to get a 1D feature vector of size 236. The extracted DF and the HF are then optimized using the SHA, and these features are then serially united to get the concatenated features vector (DF + HF). This feature vector is then considered for testing and validating the performance of the binary classifiers using 5-fold cross-validation. The experimental outcome of this study confirmed that the binary classification with the fine-tree classifier helped to achieve an accuracy of >99% for the considered chest X-ray images. This result is then compared and validated with the result of other DL methods available in the literature. This research confirmed the merit of the proposed DF + HF-based TB detection from the chest X-ray images. In the future, this scheme can be enhanced with other HF available in the literature. Furthermore, the performance of the proposed scheme can be tested and validated with other chest X-ray image datasets available in the literature.

## Data Availability

The X-ray images considered in this research work can be accessed from <https://iee-dataport.org/documents/tuberculosis-tb-chest-x-ray-database>.

## Conflicts of Interest

The authors declare that they have no conflicts of interest to report regarding the present study.

## Acknowledgments

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. 2020R1G1A1099559).

## References

- [1] V. Kumar, D. Singh, M. Damaševičius, and R. Damaševičius, "Overview of current state of research on the application of artificial intelligence techniques for COVID-19," *PeerJ Computer Science*, vol. 7, no. 1, pp. 5644–e634, 2021.
- [2] V. Chouhan, S. K. Singh, A. Khamparia et al., "A novel transfer learning based approach for pneumonia detection in chest X-ray images," *Applied Sciences*, vol. 10, no. 2, pp. 559–617, 2020.
- [3] S. J. Priya, A. J. Rani, M. S. P. Subathra, M. A. Mohammed, and N. Damaševičius, "pattern transformation based feature extraction for recognition of Parkinson's disease based on gait signals," *Diagnostics*, vol. 11, no. 8, pp. 1395–1417, 2021.
- [4] V. Rajinikanth, S. Kadry, and Y. Nam, "Convolutional-Neural-Network assisted segmentation and SVM classification of brain tumor in clinical MRI slices," *Information Technology and Control*, vol. 50, no. 2, pp. 342–356, 2021.
- [5] Z. F. Khan and S. R. Alotaibi, "Applications of artificial intelligence and big data analytics in m-health: a healthcare system perspective," *Journal of Healthcare Engineering*, pp. 1–15, 2020.
- [6] S. Vj and J. F. D., "Deep learning algorithm for COVID-19 classification using chest X-ray images," *Computational and Mathematical Methods in Medicine*, vol. 2021, no. 10, Article ID 9269173, 2021.
- [7] T. Akram, M. Attique, S. Gul et al., "A novel framework for rapid diagnosis of COVID-19 on computed tomography scans," *Pattern Analysis & Applications*, vol. 24, no. 3, pp. 951–964, 2021.
- [8] D. Singh, V. Kumar, and M. Kaur, "Densely connected convolutional networks-based COVID-19 screening model," *Applied Intelligence*, vol. 51, no. 5, pp. 3044–3051, 2021. [https://www.who.int/health-topics/tuberculosis#tab=tab\\_1](https://www.who.int/health-topics/tuberculosis#tab=tab_1).
- [9] G. Who, *Global tuberculosis report*, Glob. Tuberc. Rep, 2020.
- [10] E. Harding, "WHO global progress report on tuberculosis elimination," *The Lancet Respiratory Medicine*, vol. 8, no. 1, pp. 19–30, 2020.
- [11] A. H. Al-Timemy, R. N. Khushaba, Z. M. Mosaand, and J. Escudero, "An efficient mixture of deep and machine learning models for covid-19 and tuberculosis detection using x-ray images in resource limited settings," *Artificial Intelligence for COVID-19*, pp. 77–100, 2021.
- [12] U. Z. M. Azmi, N. A. Yusof, J. Abdullah, S. A. A. Ahmad, and F. N. M. Faudziet al, "Portable electrochemical immunosensor for detection of *Mycobacterium tuberculosis* secreted protein CFP10-ESAT6 in clinical sputum samples," *Microchimica Acta*, vol. 188, no. 1, pp. 1–11, 2021.
- [13] T. Rahman, A. Khandakar, M. A. Kadir et al., "Reliable tuberculosis detection using chest X-ray with deep learning, segmentation and visualization," *IEEE Access*, vol. 8, no. 2, pp. 191586–191601, 2020.
- [14] <https://iee-dataport.org/documents/tuberculosis-tb-chest-x-ray-database>.
- [15] V. Rajinikanth and S. Kadry, "Development of a framework for preserving the disease-evidence-information to support efficient disease diagnosis," *International Journal of Data Warehousing and Mining*, vol. 17, no. 2, pp. 63–84, 2021.
- [16] F. I. Diakogiannis, F. Waldner, P. Caccetta, and C. Wu, "ResUNet-a: a deep learning framework for semantic segmentation of remotely sensed data," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 162, no. 2, pp. 94–114, 2020.
- [17] S. Rajaraman and S. K. Antani, "Modality-specific deep learning model ensembles toward improving TB detection in chest radiographs," *IEEE Access*, vol. 8, no. 2, pp. 27318–27326, 2020.
- [18] M. H. A. Hijazi, S. Kieu Tao Hwa, A. Bade, R. Yaakob, and M. Saffree Jeffree, "Ensemble deep learning for tuberculosis detection using chest X-Ray and canny edge detected images," *IAES International Journal of Artificial Intelligence*, vol. 8, no. 4, pp. 429–442, 2019.
- [19] A. Wong, J. R. H. Lee, H. Rahmat-Khah, and A. S. A. Alaref, "TB-Net: a tailored, self-attention deep convolutional neural network design for detection of tuberculosis cases from chest X-ray images," 2021, <https://arxiv.org/abs/2104.03165>.
- [20] R. Hooda, S. Sofat, S. Kaur, A. Mittal, and F. Meriaudeau, "Deep-learning: a potential method for tuberculosis detection using chest radiography," in *Proceedings of the 2017 IEEE*

- International Conference on Signal and Image Processing Applications (ICSIPA)*, pp. 497–502, IEEE, Kuching, Malaysia, September 2017.
- [22] A. Rohilla, R. Hooda, and A. Mittal, “TB detection in chest radiograph using deep learning architecture,” *ICETETSM*, vol. 17, pp. 136–147, 2017.
- [23] Q. H. Nguyen, B. P. Nguyen, S. D. Dao, B. Unnikrishnan, and R. Dhingraet al, “Deep learning models for tuberculosis detection from chest X-ray images,” in *Proceedings of the 2019 26th International Conference on Telecommunications (ICT)*, pp. 381–385, IEEE, Hanoi, Vietnam, April 2019.
- [24] A. Afzali, F. B. Mofrad, and M. Pouladian, “Feature selection for contour-based tuberculosis detection from chest x-ray images,” in *Proceedings of the 2019 26th National and 4th International Iranian Conference on Biomedical Engineering (ICBME)*, pp. 194–198, IEEE, Tehran, Iran, November 2019.
- [25] S. Stirenko, Y. Kochura, O. Alienin, O. Rokovyi, and Y. Gordienkoet al, “Chest X-ray analysis of tuberculosis by deep learning with segmentation and augmentation,” in *Proceedings of the 2018 IEEE 38th International Conference on Electronics and Nanotechnology (ELNANO)*, pp. 422–428, IEEE, Kyiv, Ukraine, April 2018.
- [26] V. Rajinikanth, A. N. Joseph Raj, K. P. Thanaraj, and G. R. Naik, “A customized VGG19 network with concatenation of deep and handcrafted features for brain tumor detection,” *Applied Sciences*, vol. 10, no. 10, pp. 3429–3439, 2020.
- [27] L. K. Ramasamy, S. G. Padinjappurathu, S. Kadry, and R. Damaševičius, “Detection of diabetic retinopathy using a fusion of textural and ridgelet features of retinal images and sequential minimal optimization classifier,” *PeerJ Computer Science*, vol. 7, no. 1, pp. 4566–e517, 2021.
- [28] M. Wiczorek, J. Siłka, D. Połap, M. Woźniak, and R. Damasevicius, “Real-time neural network-based predictor for cov19 virus spread,” *PLoS One*, vol. 15, no. 12, p. e243217, Article ID 02431899, 2020.
- [29] O. Ronneberger, P. Fischer, and T. Brox, “U-net: convolutional networks for biomedical image segmentation,” in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, Cham, pp. 234–241, November 2015.
- [30] V. Badrinarayanan, A. Kendall, and R. Cipolla, “Segnet: a deep convolutional encoder-decoder architecture for image segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.
- [31] S. Kadry, R. Damaševičius, D. Taniar, V. Rajinikanth, and I. A. Lawal, “U-net supported segmentation of ischemic-stroke-lesion from brain MRI slices,” in *Proceedings of the 2021 Seventh International Conference on Bio Signals, Images, and Instrumentation (ICBSII)*, pp. 1–5, IEEE, Chennai, India, March 2021.
- [32] A. Gudigar, U. Raghavendra, T. Devasia et al., “Global weighted LBP based entropy features for the assessment of pulmonary hypertension,” *Pattern Recognition Letters*, vol. 125, no. 2, pp. 35–41, 2019.
- [33] L. Wei, S. X. Pan, Y. A. Nanekaran, and V. Rajinikanth, “An optimized method for skin cancer diagnosis using modified thermal exchange optimization algorithm,” *Computational and Mathematical Methods in Medicine*, vol. 2021, no. 2, pp. 1–11, 2021.
- [34] Y. Y. Wang, H. Zhang, C. H. Qiu, and S. R. Xia, “A Novel Feature Selection Method Based on Extreme Learning Machine and Fractional-Order Darwinian PSO,” *Computational intelligence and neuroscience*, vol. 2018, pp. 1–8, 2018.
- [35] K. H. Cheong, K. J. W. Tang, X. Zhao et al., “An automated skin melanoma detection system with melanoma-index based on entropy features,” *Biocybernetics and Biomedical Engineering*, vol. 41, no. 3, pp. 997–1012, 2021.
- [36] G. Dhiman and V. Kumar, “Spotted hyena optimizer: a novel bio-inspired based metaheuristic technique for engineering applications,” *Advances in Engineering Software*, vol. 114, no. 2, pp. 48–70, 2017.
- [37] G. Dhiman and V. Kumar, “Spotted hyena optimizer for solving complex and non-linear constrained engineering problems,” in *Harmony Search and Nature Inspired Optimization Algorithms*, pp. 857–867, Springer, Singapore, 2019.
- [38] A. K. Bairwa, S. Joshi, and D. Singh, “Dingo optimizer: a nature-inspired metaheuristic approach for engineering problems,” *Mathematical Problems in Engineering*, vol. 2021, no. 2, pp. 1–12, 2021.
- [39] V. Rajinikanth and N. Dey, *Magnetic Resonance Imaging: Recording, Reconstruction and Assessment*, Elsevier, Amsterdam, Netherlands, 2022.
- [40] J. Ramya, M. P. Rajakumar, and B. U. Maheswari, “Deep CNN with hybrid binary local search and particle swarm optimizer for exudates classification from fundus images,” *Journal of Digital Imaging*, vol. 35, pp. 56–67, 2022.
- [41] M. Ahmed, M. Ramzan, H. Ullah Khan et al., “Real-Time violent action recognition using key frames extraction and deep learning,” *Computers, Materials & Continua*, vol. 69, no. 2, pp. 2217–2230, 2021.
- [42] M. P. Rajakumar, R. Sonia, B. Uma Maheswari, and S. P. Karuppiyah, “Tuberculosis detection in chest X-ray using Mayfly-algorithm optimized dual-deep-learning features,” *Journal of X-Ray Science and Technology*, vol. 29, no. 6, pp. 961–974, 2021.
- [43] S. Rajesh Kannan, J. Sivakumar, and P. Ezhilarasi, “Automatic detection of COVID-19 in chest radiographs using serially concatenated deep and handcrafted features,” *Journal of X-Ray Science and Technology*, vol. 30, no. 2, pp. 231–244, 2022.
- [44] M. Hmoud Al-Adhaileh, E. Mohammed Senan, W. Alsaade et al., “Deep Learning Algorithms for Detection and Classification of Gastrointestinal Diseases,” *Complexity*, vol. 2021, Article ID 6170416, 2021.
- [45] T. Wen, Y. Du, T. Pan, C. Huang, and Z. Zhang, “A deep learning-based classification method for different frequency EEG data,” *Computational and Mathematical Methods in Medicine*, vol. 2021, pp. 1–13, 2021.
- [46] S. Sharma, S. Gupta, D. Gupta et al., “Deep Learning Model for the Automatic Classification of White Blood Cells,” *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 7384131, 2022.

## Research Article

# Optimizing Deep Learning Model for Software Cost Estimation Using Hybrid Meta-Heuristic Algorithmic Approach

Ch Anwar ul Hassan <sup>1</sup>, Muhammad Sufyan Khan,<sup>2</sup> Rizwana Irfan,<sup>3</sup> Jawaid Iqbal,<sup>1</sup> Saddam Hussain <sup>4</sup>, Syed Sajid Ullah <sup>5</sup>, Roobaea Alroobaea <sup>6</sup>, and Fazlullah Umar <sup>7</sup>

<sup>1</sup>Department of Computer Science, Capital University of Science and Technology, Islamabad 44000, Pakistan

<sup>2</sup>Department of Creative Technologies, Air University, Islamabad 44000, Pakistan

<sup>3</sup>Department of Computer Science, University of Jeddah, Jeddah, Saudi Arabia

<sup>4</sup>School of Digital Science, Universiti Brunei Darussalam, Jalan Tungku Link, Gadong, BE1410, Brunei Darussalam

<sup>5</sup>Department of Electrical and Computer Engineering, Villanova University, Villanova, PA, USA

<sup>6</sup>Department of Computer Science, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia

<sup>7</sup>Department, Khana-e-Noor University, Pol-e-Mahmood Khan, Shashdarak, 1001 Kabul, Afghanistan

Correspondence should be addressed to Saddam Hussain; saddamicup1993@gmail.com

Received 24 March 2022; Accepted 2 August 2022; Published 4 October 2022

Academic Editor: Mohamed Abdelaziz

Copyright © 2022 Ch Anwar ul Hassan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Effective software cost estimation significantly contributes to decision-making. The rising trend of using nature-inspired meta-heuristic algorithms has been seen in software cost estimation problems. The constructive cost model (COCOMO) method is a well-known regression-based algorithmic technique for estimating software costs. The limitation of the COCOMO models is that the values of these coefficients are constant for similar kinds of projects whereas, in reality, these parameters vary from one organization to another organization. Therefore, for accurate estimation, it is necessary to fine-tune the coefficients. The research community is now examining deep learning (DL) as a forward-looking solution to improve cost estimation. Although deep learning architectures provide some improvements over existing flat technologies, they also have some shortcomings, such as large training delays, over-fitting, and under-fitting. Deep learning models usually require fine-tuning to a large number of parameters. The meta-heuristic algorithm supports finding a good optimal solution at a reasonable computational cost. Additionally, heuristic approaches allow for the location of an optimum solution. So, it can be used with deep neural networks to minimize training delays. The hybrid of ant colony optimization with BAT (HACO-BA) algorithm is a hybrid optimization technique that combines the most common global optimum search technique for ant colonies (ACO) in association with one of the newest search techniques called the BAT algorithm (BA). This technology supports the solution of multivariable problems and has been applied to the optimization of a large number of engineering problems. This work will perform a two-fold assessment of algorithms: (i) comparing the efficacy of ACO, BA, and HACO-BA in optimizing COCOMO II coefficients; and (ii) using HACO-BA algorithms to optimize and improve the deep learning training process. The experimental results show that the hybrid HACO-BA performs better as compared to ACO and BA for tuning COCOMO II. HACO-BA also performs better in the optimization of DNN in terms of execution time and accuracy. The process is executed upto 100 epochs, and the accuracy achieved by the proposed DNN approach is almost 98% while NN achieved accuracy of up to 85% on the same datasets.

## 1. Introduction

Software project development includes different sets of activities, from requirements collection to testing and maintenance, which need to be executed within a specified time period and budget to achieve a reliable software product [1].

Because of the high rate of change in customer needs and rapid technological advancement, software development is more complicated than other types of engineering projects. This makes it difficult for effective software project management to achieve specific goals while adhering to a set of constraints [2]. The Standish Group report shows that only

32% of the software projects are successfully delivered on time, within the allocated budget, and have the required functionality. 44.4% did not fulfil the aforementioned requirements, and 24.4% failed; that is, they were either cancelled or completed but never used [3]. Another survey study, which was conducted among 800 senior IT managers in the US, Germany, Singapore, UK, Japan, and France, shows the same results: 62% of projects were not completed on time, 49% went over budget, and 47% required extensive maintenance [4].

Project planning is the most critical aspect of software project management and consists of a set of managerial and technical practices that can be broadly classified as the development of the project plan, execution of the project plan, and anticipating problems that may arise and preparing tentative solutions to those problems. Software cost and effort estimation come under the project-planning phase and include the process of determining how much a project will cost, how many man-hours will be required to complete the project, and how long it will take. Inaccurate estimation can result in project failure and escalation in project costs. Some of the reasons for inaccurate estimation of software projects include the following: inaccurate project goal setting, project scheduling, required development effort (capability, estimation, and availability), project budgeting, project risk management, stakeholder politics, and market pressures [5]. Effective estimates are critical in the decision-making process. Estimation must take into account both the market and the organisational perspective in order to control project costs and scope and manage the project in accordance with organisational policies. Project effort underestimation may end up in a situation where, because of a shortage of budget and time commitments, work cannot be accomplished whereas overestimation may end up with the rejection of a project proposal [6].

Several methods for solving the estimation problem are presented in the existing research literature. This effort can be divided roughly into two categories: there are two types of methods: algorithmic methods and nonalgorithmic methods [7]. Nonalgorithmic approaches rely on deduction and analogy in their estimation processes. For estimation, these models require knowledge of previously completed projects that are similar to current software projects. Previous software projects or dataset analyses are used to make estimates. Estimation techniques based on nonalgorithmic models [8] include analogy estimation, expert judgement techniques (including top-down and bottom-up estimation), learning-based methods (artificial neural network (ANN) [9–11], machine learning (ML) [12], and case-based reasoning (CBR) [13]), regression, and fuzzy logic [14].”

In the case of fuzzy logic-based methods [15, 16], although no training is required, for complex features, cost estimation becomes tedious. Furthermore, fuzzy models are difficult to use. In the case of expertise-based methods, experts determine possible costs and factors impacting the estimate. Therefore, the estimation accuracy is based on expert skill, knowledge, and experience. Learning-based techniques automatically identify trends and patterns. It gains experience and keeps improving in efficiency and accuracy. Multidimensional data of various varieties can be

handled easily through learning-based algorithms. There are also some limitations, as it requires huge data sets to train on, which should be unbiased and inclusive. Also, it takes a considerable amount of time to train and learn to fulfil a considerable amount of relevancy and accuracy. The progress of the software development process has continued to evolve over the past few decades. Therefore, most of the available data sets are heterogeneous because their sources come from various organisational projects. On the other hand, there are a large number of missing values. Therefore, the use of neural networks for high-dimensional and multiobjective data classification is a challenging task.

In the algorithmic model [17], the cost estimation is provided by a mathematical model that utilises the attributes of products, projects, and processes. These equations are derived from research and involve parameters such as function points, source lines of code (SLOC), and cost drivers (such as design methodology, risk assessments, and language dependency). Some examples of algorithmic models are the SAIC model, function point-based models, Putnam’s model, COCOMO, and SEERSEM model. COCOMO is the most extensively used regression algorithmic model for software cost estimation [18] because it allows users to adjust parameters according to the uniqueness of their projects. For estimation, COCOMO uses equations and parameters based on experience from earlier software projects. COCOMO II is an enhancement of COCOMO and is widely adopted because of its simplicity and accuracy. For effort estimation, it uses the project size (in terms of Kilo Source Line of Codes (KSLOC)) and 22 cost drivers, in which 05 scale factors and 17 effort multipliers are included. The outcome of the COCOMO II (post-architecture model) is in terms of people per month for a project. However, a prominent limitation of algorithmic models is that they are difficult to learn and require data about the current state of the project. So basically, no one method can be regarded as the best method. Therefore, it is usually recommended to combine these methods (hybrid methods) to obtain better cost estimates.

COCOMO is a parametric effort estimation model. COCOMO model coefficients play a significant role in effort estimation. The limitation of the COCOMO models is that the values of these parameter coefficients are constant for similar kinds of projects whereas, in reality, these parameters vary from one organization to another organization. Therefore, it is difficult to have a single, acceptable, and logical parametric model. The software project dataset contains data on heterogeneous projects (different project indicators in terms of scale and attributes). Therefore, in order to estimate the accuracy, the parameters need to be fine-tuned [19]. In order to overcome the limitations of COCOMO, many studies have adopted different methods to adjust the coefficients of COCOMO II and improve cost estimation. In the estimation problem, an upward trend has been seen in the use of meta-heuristic algorithms inspired by nature. Because of their unique characteristics, such as large search spaces and random selection techniques, these meta-heuristic algorithms perform well in dealing with optimization problems in various fields of interest [20].

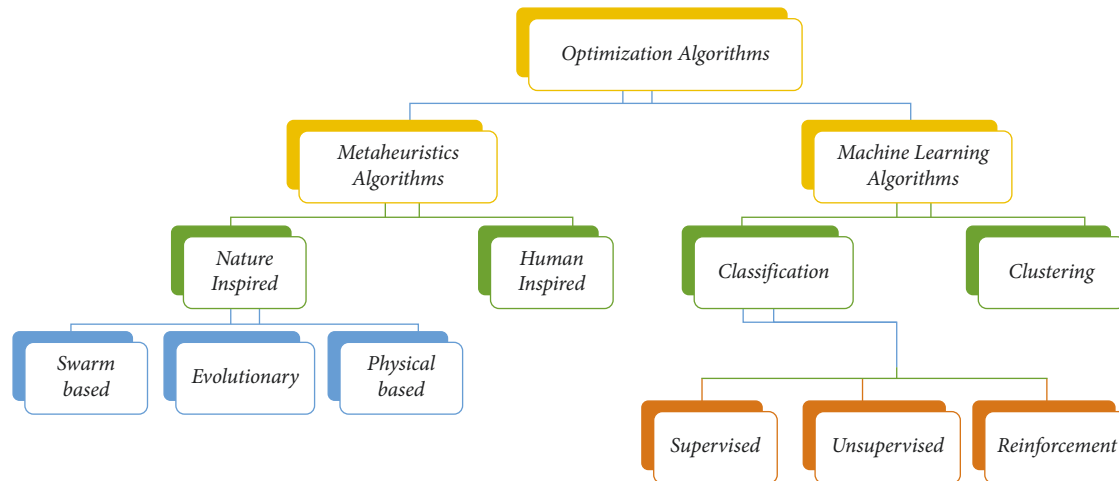


FIGURE 1: Optimization algorithms.

Deep learning (DL) is now being considered by the research community as a viable solution for improving cost estimation. Deep learning (DL) characteristics such as better feature selection and representation enable it to outperform other shallow learning techniques. Because DL makes it possible to express complex relationships between effort and cost drivers, it is a better choice for estimating software costs [21]. Although deep learning architectures offer some advantages over existing shallow technologies, they also have some drawbacks, such as long training times, overfitting, and underfitting. Meta-heuristic algorithms allow formulating the DL components into an optimization problem. The hybrid of ant colony optimization with the BAT (HACO-BA) algorithm is a hybrid optimization technique that combines the most common global optimum search technique for ant colonies (ACO) in association with one of the newest search techniques called the BAT algorithm (BA). The inclusion criteria of these algorithms are high maturity, state-of-the-art, and representative. Meta-heuristics, nature-inspired, and machine learning optimization algorithms are shown in Figure 1.

*1.1. Contribution.* RQ1: Which of the meta-heuristic algorithms has the lowest MRE, MMRE, MBRE, and PRED?

Ans: Various meta-heuristic algorithms are used for model optimization. From the literature review, we have to select different algorithms on the basis of their performance for solving various software estimation problems. Different parameters will be evaluated, for example, MRE, MMRE, MBRE, and PRED, to find out which performs better among the selected algorithms. Those algorithms are the best that have optimised values for evaluation parameters.

RQ2: whether the performance of the meta-heuristic algorithm changes or varies by changing the dataset.

Ans: The three most widely used datasets for software cost and effort estimation, that is, NASA dataset, COCOMO 81 dataset, and KEMERER dataset, will be used as input to the various models built using the meta-heuristic algorithm. The performance of all these models will be

tested and evaluated on these three datasets to see whether their performance changes or varies by changing the datasets.

RQ3: nature-inspired and meta-heuristic algorithms combined with deep learning can improve the software estimation process.

The algorithm that performs best among the other selected algorithms in terms of improvement of the evaluation parameter from RQ2 is used to tune the deep neural network (DNN). The proposed model, which is a combination of meta-heuristics and DNN, is tested and evaluated for fast and efficient results.

RQ4: whether the performance of a meta-heuristic-based deep learning algorithm changes by changing the dataset.

Different datasets are chosen from the state of the art. These datasets are used as input to the proposed meta-heuristic-based DNN. Performance is tested and evaluated for this model by changing the datasets.

Whether metaheuristic-based deep learning algorithms perform better than the NN-based approach for software estimation.

Ans: Neural network (NN)-based approaches for software estimation are widely used by the research community. However, there are various pros and cons to using NN techniques. Solving various optimization problems with DNN is the most popular topic among researchers. So we have to check whether the meta-heuristic with DNN performs best when compared to the NN model. Table 1 describes the notation guide for each algorithm as well as additional abbreviations.

The rest of the article is formatted as Section 2 demonstrates related work and a subsection of research gaps, Section 3 contains a problem statement, Section 4 describes the proposed methodology, and results are discussed in Section 5.

## 2. Related Work

Many researchers and practitioners around the world use different methods to improve software estimation [17]. The



TABLE 1: Notation guide.

Notations	Abbreviation
ML	Machine learning
ACO	Ant colony optimization
BA	BAT algorithm
MMRE	Mean magnitude relative error
NN	Neural network
DL	Deep learning
COCOMO	COnstructive COSt Mode
BCO	Bee colony optimization
PSO	Partial swarm optimization (PSO)
COCOMO	COnstructive COSt Model
SLOC	Source lines of code
RQ	Research questions
MAR	Mean absolute residual
RBFNN	Radial basis function NN
RF	Random forest
COA	Chaos optimization algorithm
ABC	Artificial bee colony
OPSO	Optimized particle swarm optimization
MBRE	Mean balanced relative error
INGPS	IdeNtitybased generalised proxy signcryption

existing literature in the field of software estimation has proposed various techniques to evaluate the accuracy of predictive models. The most popular among the existing literature is the mean magnitude of relative error (MMRE), based upon mean relative error as shown in the following equation:

$$\text{MRE} = \frac{\text{actualefforts} - \text{predictedefforts}}{\text{actualefforts}}. \quad (1)$$

The lower the MMRE value, the closer the predicted estimated value is to the actual estimate, and vice versa. There is work being done to review existing studies on software cost estimation [19]. Various meta-heuristic techniques for software cost estimation have been implemented over the last decade. There is work that computes the effectiveness of meta heuristics algorithms [17, 20, 22–24] related to the optimization of software cost estimation. For example, in the existing literature, genetic algorithm (GA) [23], hybrid GA [24], ants colony optimization (ACO) [25] algorithm, and firefly algorithm (FA) [26] improved cost estimation. Moreover, existing literature also demonstrates the effectiveness of meta-heuristic algorithms in terms of optimizing the parameters of COCOMO [25–29]. There is work to optimize COCOMO II coefficients using hybrid algorithms. A hybrid method is [30–36] the combination of several methods that can be derived from algorithmic or nonalgorithmic techniques. For example, the author of [34] uses the artificial bee colony and the genetic algorithm for optimization.

Bee colony optimization (BCO) [22] is a subclass of swarm intelligence that has been effectively used in a variety of engineering applications, including software estimation. By using BCO, COCOMO parameters have been optimised. In the proposed technique, artificial agents are produced by analogy with bees. Better results are obtained as compared to other models such as the Baily-Basil and Halsted models.

Puri and Kaur [20] discussed various meta-heuristic techniques that are used for software cost and effort estimation. BCO works on the natural phenomenon of getting food from bees. It has two stages: moving forward and moving backward. FA is based on firefly flashing characteristics, and human opinion dynamics (HOD) is based upon the human creative problem-solving process to solve complex optimization problems.

The parameters of the basic COCOMO model are optimised using a simplified GA technique in [23]. We use the NASA software project dataset as a starting point. According to empirical evidence, the basic COCOMO model produces a superior actual estimate. The authors of [24] proposed the whale-crow optimization (WCO) algorithm, which is a combination of the whale (WOA) and crow search (CSA) optimization algorithms. The main objective of the WCO approach is to find an optimal regression coefficient to build an optimal regression model. The performance evaluation of the proposed scheme is calculated using four datasets of software estimation. MMRE is analysed, and a reduction is seen, which proves that WCO performs well in both the linear regression model and the kernel regression model. The authors of [26] proposed the firefly algorithm (FA) as a meta-heuristic optimization technique for optimizing three COCOMO-based model parameters. Among these three models, the basic COCOMO model and the other two models are proposed in the state-of-the-art as an extension of the basic COCOMO model.

Jafari and Ziaaddini [27] analysed the effectiveness of the harmony search algorithm (HSA). Using the NASA dataset, the work demonstrates a significant reduction of MMRE as compared to basic COCOMO. To estimate software reliability issues, the partial swarm intelligence (PSO) technique and ant colony optimization (ACO) are used [28]. In [29], the BAT algorithm is presented to improve the software estimation accuracy. The effectiveness of the BAT algorithm is compared with grey relational analysis (GRA). GRA delivers an effective solution for the complex interrelationships between multiple response parameters. The results show that GRA performs better in terms of error rate.

A new hybrid model, BATGSA [30], which is based on two meta-heuristic algorithms, gravitational search, and the BAT Algorithm, is proposed to achieve better software estimation. The BAT algorithm uses the random walk of BATs to determine the hunting and routing behaviour of bats in the exploration phase and uses the gravitational effect of GSA to further improve and speed up the search speed of BATs. Four NASA datasets are used for analysis, downloaded from the promise repository. The minimization of errors is obtained by comparing the COCOMO against the hybrid BATGSA algorithm.

The work presented in [31] demonstrates that software cost estimation can be improved by using a hybrid approach based on Tabu search and genetic algorithms. Furthermore, the hybrid model is used to tune the COCOMO.

In [32], three meta-heuristic optimization algorithms are implied synthetically in order to refine the COCOMO model: partial swarm optimization (PSO), invasive weed optimization (WOA), and genetic algorithm (GA). The

dataset is divided into two groups: train and test datasets. Furthermore, the dataset is further divided based on the type of projects. COCOMO parameters are tuned using meta-heuristic algorithms, and the new cost drivers are tested using a testing dataset.

In [33], a hybrid approach based on ant colony (ACO) and chaos optimization algorithm (COA) is presented. In the first phase, the dataset is classified into two parts according to project type; one is used for training the model, and the second is used to test that trained model. Each algorithm (ACO, hybrid ACO-COA) is applied with its own function in the first phase. Next, the most optimal solutions obtained for estimation from the proposed model are then tested using testing data. In the result section, a comparison is made between COCOMO, ACO, and the hybrid ACO-COA model.

In [34], the author proposed a hybrid model based on genetic algorithm (GA) and artificial bee colony (ABC) schemes for optimization of estimation method parameters according to project size. In the artificial bee colony (ABC) algorithm, each bee presents a solution to the problem. In every iteration, the most graceful bee solution is chosen, and their fitness is calculated. The bees whose performance is low are replaced with new bees. The new bee population is generated using GA crossover, selection, and mutation. Finally, the most optimal solution is found. The results demonstrate that GA, ABC, and the hybrid model of ABC with GA performed well (fewer MRE error values) compared to the COCOMO model. Moreover, the hybrid model demonstrates better convergence as compared to the GA and ABC algorithms.

In [35], the author uses a hybrid model formed on PSO and DE algorithms to deliver a more comprehensive and efficient estimate. With incomplete and ambiguous input data, the hybrid model works well, and it can operate reliably in software estimation. In the proposed hybrid model, the accuracy of PRED (25) increased 1.34 times.

In [36], a hybrid model based on the cuckoo search (CS) and harmony search (HS) algorithms is used for optimizing COCOMO-II coefficients. The proposed CSHS has two stages. CS at the first stage is used for finding the initial optimal solution for local and global search. In the second, a new harmony vector or solution is generated using the HS strategy, which is compared with the global best solution found in the first stage. If it has better fitness, then the global best is replaced. This procedure is continued until the given iterations are completed. The proposed technique is applied to NASA 93 datasets. The aim is to achieve an estimation value close to the actual value.

ML is a method that trains computing systems to improve itself by learning from previous data available. ML programs work by constructing a prediction model from a set of previously available training data, and this step is followed by the data-driven predictions [37].

Categorically, analysis was performed in [21] on 4 neural network models: (1) general regression NN (GRNN), (2) multilayer perceptron (MLP), (3) cascade correlation neural network (CCNN), (4) radial basis function NN (RBFNN), and mean absolute residual (MAR) were the criteria used for performance evaluation. Four inputs have been assigned to

each model, including (1) development platform, (2) software size, (3) resource level, and (4) language type. Five datasets mined from the ISBSG are used. The output of the model was software effort.

In [38], an empirical study was conducted by using the random forest (RF) method for software effort estimation. In RF, first, we investigate the number of trees affected and then evaluate the number of selected attributes for growing trees. The outcomes show that the estimation accuracy is very sensitive to these parameters.

In addition, the survey conceded that we optimize the RF model by selecting the best values for these two parameters. We compare the performance of the enhanced RF model with the performance of the classical regression tree (RT) by using the (70–30) hold-out validation method and using three COCOMO, IBSSG, and Tuketuku data sets.”

Some highly mature and popular ML algorithms, such as support vector machines, regression, decision trees, RF, Bayesian inference, ANN [39], and feature selection, are an important process during training a model because model efficiency depends on selected variables. It is very important to choose features that have a significant influence on the prediction model [40].

Barmpalexis et al. [41] accelerate the neural network training process by using optimized particle swarm optimization (OPSO). The main function of OPSO is to enhance the free PSO parameters by having a new swarm within a swarm. The aim is to build a quantitative model by applying the OPSO technique to neural network training. This method yields the parameter combinations needed to improve the overall performance of the optimization process.

In [42], the author proposed a nonalgorithmic method for estimating software development effort. This article discusses the integration of wavelet neural networks (WNN) and meta-heuristic methods for estimating software development effort (SDEE). The technologies used here are WNN with the firefly algorithm and the BAT algorithm. As the activation function in WNN, two wavelet function variants—Morlet and Gaussian—are used. It has been discovered that using WNN with the firefly and BAT algorithms (FA and BA) produces better results than using simple WNN without any meta-heuristics. According to the author’s experimental results, the WM technique performs the worst across all four data sets. However, combining meta-heuristics with WNN yields significantly better results.

In [43], the author conducted an exploratory longitudinal case study. Data collection was conducted through semistructured interviews and archival research. The two-stage estimation process, which reestimates in the analysis stage, improves the effort estimation accuracy.

Underestimation is the main trend in software evaluation, and less mature teams experience greater work overspending.

Some of the most common challenges are solved in large-scale agile software development. In order to improve the effort estimation, the team maturity, distribution, and demand size and priority need to be considered.

Emary et al. [44] proposed a modified grey wolf optimization (GWO) that utilises the reinforcement learning

rules or principles by integrating them with neural networks in order to improve the model performance. The combination of GWO with a neural network forms experienced GWO (EGWO). The performance of experienced GWO is measured by finding the optimal weights of the neural network and choosing a subset of related features (predictors, variables) for use in model building.

The author in [45] presents a comprehensive dataset for the story points-based estimation. 23,313 issues from 16 open-source projects are addressed. A prediction model using DNN is also proposed for the estimation of the story points.

In [46], the author proposed a new technique of deep learning modified neural network using the cuckoo search algorithm for initialising the weight of the network and applying HPSO to obtain a better classification of various parameters of the dataset. A neural network is qualified as a “deep network” when there are more than three layers. The deep modified neural network (MNN) classifier is comprised of neurons with weights in addition to biases. Also, the deep MNN classifier consists of different sorts of layers: convolution, pooling, as well as a fully connected layer. The cuckoo search algorithm (CSA) is used to initialise the weight of the network. The input of this deep MNN is the effort multiplier, that is, the software development of the COCOMO dataset, database size, exponent value, constant value, etc. This step is carried out through an optimization process, which is executed by using hybrid particle swarm optimization (HPSO) with genetic operators. (Crossover, explicitly, and also mutation of genetic algorithms). While choosing the NN weights, which help to enhance the classification of the model, HPSO is used to attain better classification outcomes. Finally, the proposed deep MNN is evaluated on different performance parameters: relative error (RE), magnitude of relative error (MRE), mean-MRE (MMRE), mean balanced relative error (MBRE), and also percentage of prediction (PRED) and compared with traditional NN. The experiment shows better results in all features. The execution time for effort estimation is also compared. When 10 instances are considered, the proposed deep MNN shows a decrease in execution time compared with the traditional NN. On the other hand, there is also a limitation in the proposed technique; that is, when the instances were increased to 50, the results show that the proposed method takes more time.

A hybrid approach is introduced in [11], which is divided into two sections. First, the author applies the GA for feature optimization. To obtain the desired results, the total population is divided into several subpopulations and applies computation for each population, which includes designing of chromosomes and calculation of fitness functions. In the second section, an improved DNN for classification is proposed. As we know, neural networks show significant performance in terms of classification. However, neural networks cannot classify multiobjective functions or high-dimensional data, so the paper proposes an enhanced DNN model with sparse auto-encoders to overcome this limitation. We use the proposed technique to learn the feature

pattern. The adaptive auto-encoders are used in conjunction with the denoising model to produce better results for specific software features. The MATLAB tool is used for experiments in various scenarios that are performed for software defect prediction. The proposed technique’s performance is assessed using data sets KC1 and CM1. A comparative study reveals that the proposed technique outperforms the experimental scenario without optimization in all four scenarios created during the experiments. A brief summary of estimation methods and their limitations is shown in Table 2.

For categorised problems, the deep belief network (DBN) is an excellent machine learning technology. The traditional DBN, on the other hand, does not function well for unbalanced data classification because it assumes that each class has the same cost. Cost-sensitive approaches are employed to overcome this issue, which attaches varying misclassification costs to different classes without affecting the actual data sample distribution. The author [47] proposes an evolutionary cost-sensitive deep belief network (ECS-DBN) model in which he first optimises the misclassification costs using optimization algorithms that automatically update their corresponding parameters and then applies them to the deep belief network. The author demonstrated that ECS-DBN outperforms other competing techniques significantly. The suggested ECS-DBN improves DBN by applying cost-sensitive learning techniques. The adaptive differential evolution approach is utilised in practise to find the misclassification cost and solve the unknown misclassification cost [48–50]. In [51], the grey wolf algorithm (GWO), the strawberry algorithm (SBA), and the harmony search algorithm (HSA) were tested on MRE and MMRE parameters using the NASA dataset.

The author proposed multiple approaches to make the software secure and reduce future efforts for maintenance. In [52], the author proposed a lightweight identity-based signature scheme for content poisoning mitigation in named data networking with the Internet of things. A secure identity-based generalised proxy signcryption (IBGPS) scheme that is lightweight and provable for the industrial Internet of Things (IIoT) is proposed in [53]. For the Internet of Things-enabled smart grid, CBSRE is a lightweight and formally secure certificate-based signcryption with proxy reencryption. For the named data networking-enabled Internet of things, a lightweight heterogeneous generalised signcryption (hgsc) scheme, securing the NDN-based Internet of Health Things with a low-cost encryption scheme, is proposed in [54, 55]. Machine and deep learning approaches are widely used in different areas of life. The author proposed an intrusion detection system for IoT based on deep learning and a modified reptile search algorithm in [56] and a modified Aquila optimizer for forecasting oil production in [57]. In [58], author forecasted the wind power using the marine predator algorithm and mutation operators for wind power forecasting to evaluate the performance of meta heuristic approach.

TABLE 2: Estimation method and limitations.

Estimation method	Limitations
Estimation by analogy	Subjective selection of correlation standards and dispute identification process (confidence level) Requires analogous project for comparison from historical data from database These analogous projects are rarely available in software development
Decomposition and bottom-up (WBS-based)	It may be time-consuming for large or even medium-sized projects High risk of ignoring system-related tasks such as testing, integration, and configuration is high This method may lead to underestimation due to lack of project information at early stage
Parametric models (SLIM, SEERSEM)	Usually does not take into account the project team's skill set specific to the organization's software and project management culture Modern methods of code reuse, code less programming, and various agile development methods for software development may not be feasible Highly dependent on programming language
Expert estimation (Delphi, PERT, planning poker)	These methods rely on the experience, knowledge, and perception of experts, and there may be deviations or biased, which often lead to overestimation or underestimation All the factors used by experts in the estimation process are unable to justify and quantify
Size-based estimation models (use case, FPA, sTory points)	Requires trained personnel which is not easily available High effort and cost is required for the application of large projects Due to limited information, using this method in the early stages of a project may result in inaccurate estimates

TABLE 3: Comparison between existing approaches.

Refer ence	DL/ML/ ANN	Meta-heuristic algorithm	Dataset	Evaluation parameter	Contributions
[42]	NN	Fiery algorithm, BAT algorithm	COCOMO81, NASA, MAXWELL, China	MRE, MMRE, pred, MDMRE	Hybrid model for effort estimation
[45]	ANN	Firefly	COCOMO81, NASA, MAXWELL, China	MMRE, MdMRE, PRED	Hybrid model for cost estimation
[46]	DNN	Cuckoo, hybrid PSO	COCOMO	RE, MRE, MMRE, MARE, PRED, execution time	Hybrid model for cost estimation
[11]	DNN	GA	KC1, KC2, CM1, PC1, JM1	Accuracy, precision, F-score, recall, sensitivity	Defect prediction
[47]	DL	Evolutionary algorithm	KEEL dataset repository	Accuracy, G-mean, precision, F-score, computational time	Hybrid of DBN and ADE for imbalanced classification
[48]	NN	GA, PSO	N/A	Survey	Possibility to apply on DL
[49]	NN	Cuckoo	COCOMO	MMRE, standard deviation	Improve cocomo
[50]	ANN	Cuckoo	COCOMO81, NASA	MMRE, PRED, computational time	Hybrid model
[51]	GWO, HSA	SBA	NASA	MRE, MMRE	Hybrid model

### 3. Algorithmic Approaches

Meta-heuristic algorithms and nature-inspired algorithmic approaches are discussed in this section.

*3.1. Meta-Heuristic Algorithms.* Nature influenced meta-heuristic algorithms and nature-inspired algorithms. Natural biological systems, evolution, human activities, animal group behaviours, and other factors can inspire algorithms, such as the biological human brain-inspired artificial neural network [1], the genetic algorithm stimulated by evolutionary theory [2], and Dujuan. The cuckoo search algorithm (CSA) was inspired by the cuckoo's birth behaviour [3], whereas the grey wolf optimization (GWO) was inspired by the grey's aggressive behaviour [4].

It is found that these nature-inspired algorithms are more effective and efficient than traditional algorithms in solving real-world optimization problems because they can effectively deal with highly complex and nonlinear problems, especially in the fields of science and engineering [22]. Meta-heuristic is defined as an iterative method, which explores and uses search space to guide lower-level heuristics by intelligently combining different concepts. They are inspired by observing phenomena that occur in nature. The summary of the used meta heuristics techniques is presented in Table 3.

*3.1.1. Ant Colony Optimization.* Marco Dorigo was first introduced in 1992 as a multiagent solution for optimization problems. When an ant moves, it deposits pheromone (in

varying amounts) on the ground and uses the smell of this substance to determine its path. The colony's other members follow the path to find food and return to the nest in the same manner. Ants begin their search for food sources by randomly exploring the area around their nest [25]. When ants find a food source, they evaluate the quality of the food and bring a small amount back to their nest. The flow chart of the ant colony optimization algorithm is shown in Figure 2.

$$\tau_{i,j}(t+1) = \tau_{i,j}(t) + \sum_{k=1}^{n_k} \delta \tau_{i,j}^k(t). \quad (2)$$

Ants communicate via indirect channels and coordinate their activities in a hidden state by doing changes in the surrounding environment as shown in equation (2). ACO works on the principle of indirect artificial communication to match societies of artificial agents. The steps of ant colony optimization algorithms are shown in Algorithm 1.

**3.1.2. BAT Algorithm.** The BAT algorithm (BA) [30, 42] was proposed by Xin-She Yang and is influenced by mini-BAT echolocation behaviour. BATs use this behaviour to direct and assist them in flight and hunting. BATs can not only move but also discern between obstructions and bug forms, even in complete darkness, thanks to their incredible orientation mechanism. The flow chart of the BAT algorithm is shown in Figure 3.

The position of each BAT in the search space is defined by  $x_k^t$ , frequency  $f$ , velocity  $v_k^t$ , loudness  $A_k^t$ , and transmitted pulse rate  $r_k^t$  in this algorithm. The velocity and position  $k^{th}$  of the BAT at time  $t$  are calculated using the Equations 3rd, 4th, and 5th.

$$\begin{aligned} f_k &= f_{\min} + (f_{\max} - f_{\min})\beta, \\ v_k^t &= v_k^{t-1} + (x_k^{t-1} - x_k^t)f_k, \\ x_k^t &= x_k^{t-1} + v_k^t. \end{aligned} \quad (3)$$

Among them,  $f_k$  is the frequency of the sound waves emitted by  $k^{th}$  BATs;  $f_{\min}$  and  $f_{\max}$  are the minimum and maximum sound waves frequencies, respectively;  $\beta$  is composed of uniform which distributes the random number generated by  $[0, 1]$ . The velocities of  $k^{th}$  BAT are  $v_k^t$  and  $v_k^{t-1}$  at time  $t$  and time  $(t-1)$ , and  $x_k^t$  represents the current global optimal position of the BAT.

For local search, each bat the position is measured using equation (6). Each BAT local random walk is calculated by using the following equation:

$$X_{\text{new}} = X_{\text{old}} + \delta \dot{L}(t), \quad (4)$$

where  $\delta$  is a random number produced uniformly distributed on the interval  $[-1, 1]$ ,  $X_{\text{old}}$  is a solution arbitrarily selected from the current optimal solution, in the iteration of  $i^{th}$ ,  $\dot{L}$  is the normal uproar of all BATs. The steps of the BAT algorithm are shown in Algorithm 2.

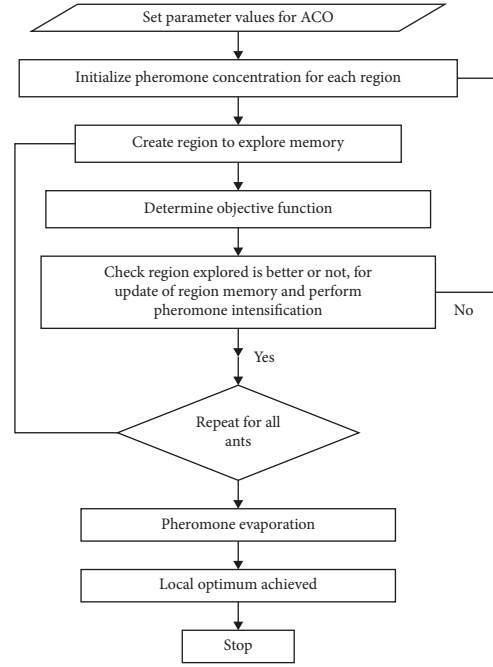
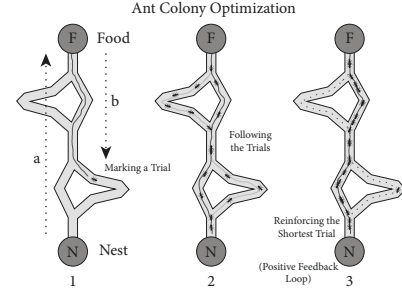


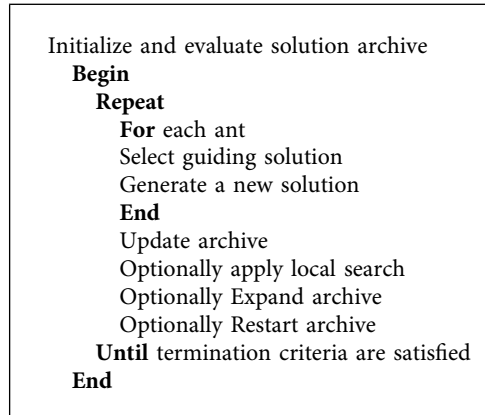
FIGURE 2: Ant colony algorithm.

**3.1.3. Hybrid of Ant Colony Optimization with BAT (HACO-BA).** The hybrid of ant colony optimization with BAT (HACO-BA) algorithm is a hybrid optimization technique that combines the most common global optimum search technique for ant colonies (ACO) in association with one of the newest search techniques called the BAT algorithm 3 (BA).

**3.2. Performance Analysis and Evaluation.** In this section, we discuss about the performance analysis, experimental setup, datasets selection, evaluation metrics, and experimental results.

**3.2.1. Performance Analysis.** In the experimental section, we will compare the performance of 6 meta-heuristic algorithms, that is, GWO, GA, strawberry (SBA), cuckoo search, particle swarm (PSO), and ant colony optimization (ACO) [23, 25, 28, 34, 35, 59]) that use meta-heuristic algorithms in terms of effort and cost estimation with each other and with COCOMO model.

In 1981, Barry Boehm proposed the constructive cost model (COCOMO). It is the most often used algorithmic or parametric model. The model parameters and equations are generated from historical projects for estimation. The model



ALGORITHM 1: Ant colony optimization (ACO).

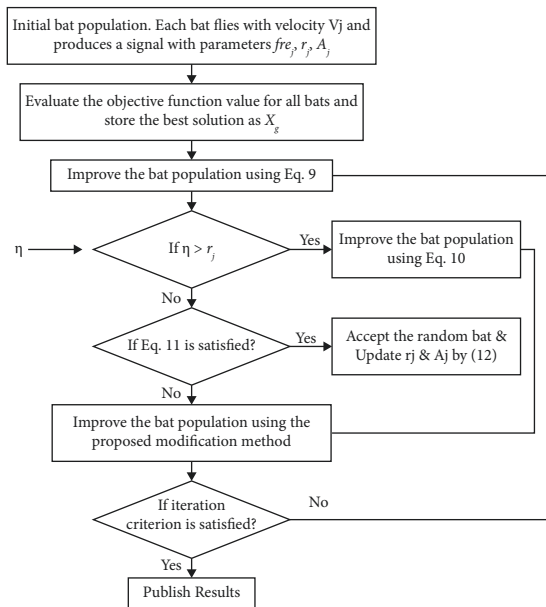
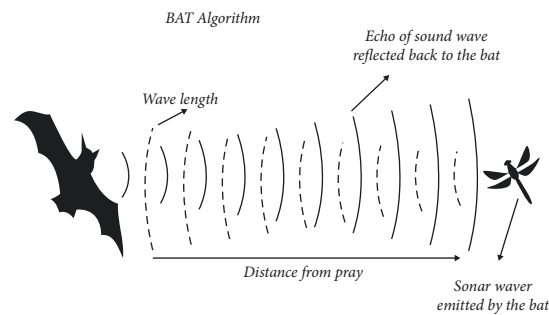


FIGURE 3: BAT algorithm.

is used to estimate the project’s size, the amount of effort necessary, and the project’s cost. We must first construct a set of criteria to quantify this. In this model, we use functional points (FP) and lines of code (LOC) to compute the required efforts. The “person month” unit is used to estimate effort in this method, which is equivalent to a single person’s month efforts. COCOMO models are divided into two categories.

- (i) COCOMO I
- (ii) COCOMO II

COCOMO I (also known as COCOMO 81) and COCOMO II (COCOMO 2000) were released in 1981 and 1995, respectively. It was, however, published in the year 2000. COCOMO I is separated into three levels of difficulty:

```

For a given objective function:  $Obj(x_i)$ , where  $i = (1, \dots, n)$ 
Initialize the BAT population with the corresponding attributes: velocity  $vel_i$ , position  $(x_i)$  and pulse frequency  $(Q_i)$ 
Initialization features include loudness  $loud_i$ , maximum number of iterations ( $Iter_{max}$ ) and acceleration rate  $rate_i$ 
  Repeat step 4 to 15 for every iteration in ( $Iter_{max}$ )
  Repeat step 5 to 13 for every single BAT corresponding to  $bat_i$ 
  Evaluate equations (1)–(3) to produce a new solutions
  set
  if (random >  $r_1$ ) then
    Generate a local solution around one of the chosen best solutions
  end if
  if (random <  $loud_i$ ) ( $fitness(x_i < fitness(x_{GlobalBest}))$ ) then
    Update  $x_{GlobalBest}$  and  $fitness_{Best}$ 
    Increase  $rate_i$  and reduce  $loud_i$  (equations (3) and (6))
  end if
Rank the BATs and find the  $Global_{Best}$ 
=0

```

ALGORITHM 2: BAT algorithm (BA).

```

Initialize and evaluate solution archive
Begin
For a given objective function:  $Obj(x_i)$ , where  $i = (1, \dots, n)$ 
Initialize the BAT population with the corresponding attributes: velocity  $vel_i$ , position  $(x_i)$  and pulse frequency  $(Q_i)$ 
Initialization features include loudness  $loud_i$ , maximum number of iterations ( $Iter_{max}$ ) and acceleration rate  $rate_i$ 
  Repeat step 4 to 15 for every iteration in ( $Iter_{max}$ )
  Repeat step 5 to 13 for every single BAT corresponding to  $bat_i$ 
  Evaluate equations (1)–(3) to produce a new solutions set
  if (random >  $r_1$ ) then
    Generate a local solution around one of the chosen best solutions
  end if
  if (random <  $loud_i$ ) ( $fitness(x_i < fitness(x_{GlobalBest}))$ ) then
    Update  $x_{GlobalBest}$  and  $fitness_{Best}$ 
    Increase  $rate_i$  and reduce  $loud_i$  (equations (3) and (6))
  end if
Rank the BATs and find the  $Global_{Best}$ 
  Update archive
  Optionally apply local search
  Optionally Expand archive
  Optionally Restart archive
Until termination criteria are satisfied
End =0

```

ALGORITHM 3: Hybrid ant colony and BAT algorithm (HACO-BA).

basic, intermediate, and advanced [1]. The formulas utilised to calculate the estimations change across these methods. The most widely utilised methods are basic and intermediate; these approaches are further divided into three sections or measurements. These are organic, semidetached, and embedded, according to the projects featured in it [2].

Basic COCOMO  $E$ , organic COCOMO  $E_{B-O}$ , semi-detached COCOMO  $E_{B-S}$ , and embedded COCOMO  $E_{B-E}$  formulas are shown in the following equations, respectively.

$$E = a(\text{size})^b, \quad (5)$$

$$E_{B-O} = 2.4 * (LOC)^{1.05}, \quad (6)$$

$$E_{B-S} = 3.0 * (LOC)^{1.12}, \quad (7)$$

$$E_{B-E} = 3.6 * (LOC)^{1.20}. \quad (8)$$

These are the fundamental COCOMO equations. KLOC represents the number of code lines and project size in these

equations. The COCOMO coefficients “ $a$ ” and “ $b$ ,” as well as the value of “ $E$ ,” describe the required efforts.

**3.2.2. Experimental Setup.** This stage is building a model in MATLAB software to estimate effort using six algorithms selected for their excellent performance in diverse optimization situations [25–27]. Because its basic data element is a matrix, and its capability can be easily increased by utilising multiple toolboxes, using MATLAB software provides significant advantages. On three publicly available datasets retrieved from the promise repository, various tests are carried out using cutting-edge algorithms.

**3.2.3. Datasets Selection.** Effort multipliers are taken as input to the models, from the following defined datasets. These effort multipliers are categorized into three groups, which are as follows:

- (i) Positively correlated to additional effort
- (ii) Negatively correlated to additional effort
- (iii) Containing just schedule information

(i) *NASA*. The NASA dataset was obtained from the promise software engineering repository, which can be used publicly to improve software cost estimation methods. It includes 93 software project information, which has been recorded for many years from several NASA centers. Dataset contains 15 effort multipliers and 5 scaling factors, which have different values in each software project.

(ii) *COCOMO 81*. This dataset is also known as COCOMO 81 which is publicly available on promise software engineering repository. This repository’s software project data are stored in the COCOMO software cost model, which calculates the amount of effort required to develop software projects in a calendar month. It also includes a standard effort multiplier.

(iii) *KEMERER*. KEMMER dataset is measured in KLOC. It is used in many machine learning applications for software engineering which has 8 attributes. To compare it with Nasa and COCOMO dataset, which has 15 attributes, we assume the rest of attribute value as normal.

**3.2.4. Evaluation Matrices.** Many researchers and practitioners have optimised the effort estimation technique to assist the accuracy under various estimation standards. We implement the following standards to compare and evaluate the accuracy of the effort estimation model.

(i) *MRE*. One of the common criteria for evaluating the effort estimation process is the magnitude of relative error (MRE), which is computed using the following equation:

$$MRE = \left| \left( \frac{\text{Actual} - \text{Estimate}}{\text{Actual}} \right) \right|. \quad (9)$$

(ii) *MMRE*. The MRE value is calculated from the dataset for each software item, while the mean magnitude of relative error (MMRE) calculates the average of  $N$  number of projects, as defined in the following equation:

$$MMRE = \frac{1}{N} \sum_{i=1}^N MRE.i. \quad (10)$$

(iii) *MBRE*. MBRE is another measure that is commonly used to evaluate effort models. In recent years, it has been the average value of balanced relative error (MBRE) in software estimation research [12]. MBRE, in particular, is a useful evaluation standard because, as a balanced symmetric error measure, it penalizes both underestimation and overestimation at the same level and better handles the outline. MBRE is calculated in the following equation:

$$MBRE = \frac{1}{N} \sum_{i=1}^N \frac{|\text{Actual} - \text{Estimate}|}{\min(\text{Estimate} - \text{Actual})}. \quad (11)$$

(iv) *PRED*. The other most common metric is PRED( $I$ ), which represents all projects with an MRE percentage less than or equal to the  $I$  value. This standard, which is commonly used in the literature, is the proportion of projects completed with a given level of accuracy. In equation (12),  $\text{pred}(x)$  is defined.

$$\text{Pred}(x) = \frac{k}{N}, \quad (12)$$

where  $k$  denotes the number of projects whose MRE is equal to or less than  $x$ , and  $N$  denotes the total number of projects. The most common value of  $x$  is 0.25, which is also used in this study.  $\text{Pred}(0.25)$  denotes the percentage of projects with MREs equal to or less than 25.

Estimation refers to the estimated value of the predicted efforts, actual refers to the actual workload or effort required to complete the project, and  $N$  denotes the number of projects.

**3.2.5. Experimental Result.** In this section, optimization is carried out on all modes discussed above; it is performed on the NASA dataset, on semidetached mode, and on embedded projects, and the average of each mode is taken. The model also receives input from the COCOMO 81 dataset. During experiments, the KEMERER dataset is also used.

(i) *Experiment 1*. Optimization is performed on the NASA dataset by using different nature-inspired algorithms. As results are shown in Figure 4, the MMRE value is decreased by using the meta-heuristic algorithms as compared to the basic COCOMO parametric model. All the datasets are divided into three folds, and then, the average is plotted on the graph, as shown in Figure 4. MMRE is decreased by using ACO, BAT, and HACO-BA algorithmic approaches. We noticed that the hybrid algorithmic approach HACO-BA



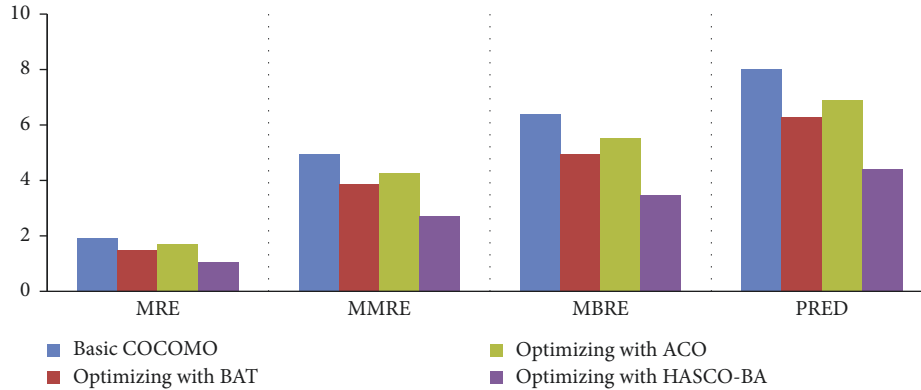


FIGURE 4: Comparison of NASA dataset.

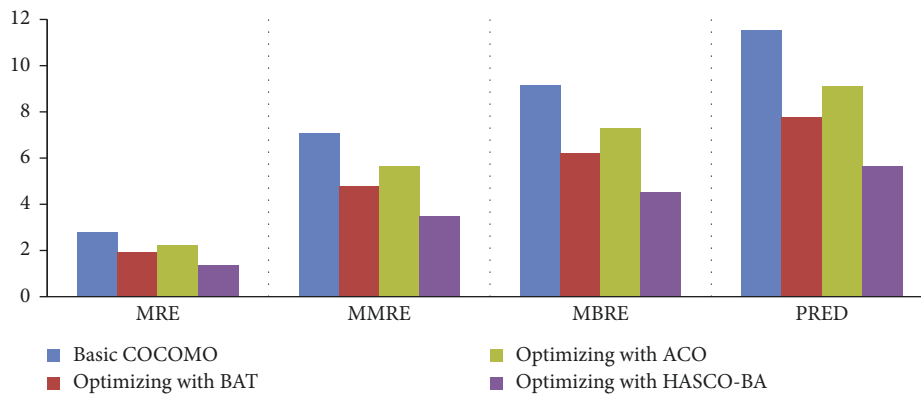


FIGURE 5: Comparison of COCOMO 81 dataset.

shows a significant decrease as compared to other optimizing algorithms.

(ii) *Experiment 2.* The experimental results on the COCOMO 81 dataset reveal that the maximum value of MMRE is 7 as shown in Figure 5 as compared to the NASA dataset and the maximum value of MMRE is 5 as shown in Figure 5. Experimental results also show a decline in MMRE and other evaluation parameters while using the hybrid HACO-BA approach.

(iii) *Experiment 3.* KEMERER dataset is also used to evaluate the performance of applied meta-heuristics approaches. In KEMERER dataset, we have 8 attributes, and to balance it with the above dataset, the rest of 7 attributes is assumed as normal whose value is equal to 1. Due to this, the decrease in MMRE is less as compared to NASA and COCOMO 81 datasets. On KEMERER, the dataset hybrid approach HACO-BA performs well compared to other meta-heuristic algorithms as shown in Figure 6.

In Table 4, valuation parameter values of the optimization models such as Basic COCOMO, BAT, ACO, and HACO-BA on different datasets are listed. These approaches are evaluated on MRE, MMRE, MBRE, PRED evaluation parameters using three different datasets; NASA, COCOMO, and KEMERER are listed.

#### 4. Proposed Software Estimation Scheme Based on Hybrid Meta-Heuristic and Deep Learning Model

In the proposed system, we used the deep learning model. Deep learning is a type of artificial neural network architecture (ANN). ANN represents a significant early breakthrough in the field of artificial intelligence. The ANN model is exceptionally dynamic in solving complex problems in various machine learning application areas [11] in the real world, such as health, agriculture, finance, and automobile industry. At the moment, ANN in single, hybrid, or ensemble form is still an active research area [12], and its role in autonomous vehicles is expected to receive more attention in the future. ANN, on the other hand, is trained using backpropagation algorithms and has some limitations, such as falling into local minima and overfitting training data. As a result, many researchers advocate using nature-inspired algorithms to train ANNs to avoid challenges. For example, GA [24], ABC [34], CSA [36], and particle swarm optimization (PSO) [41] were used to train ANN and were found to be superior to the back propagation algorithm in terms of avoiding the local minima problem.

As stated above, deep learning is an ANN architecture with logical node weight updates and activation functions. Deep learning models and extracting high-level abstractions from large-scale data sets are useful when providing large-

TABLE 4: Parameter evaluation using different datasets.

Optimization models	NASA				COCOMO				KEMERER			
	MRE	MMRE	MBRE	PRED	MRE	MMRE	MBRE	PRED	MRE	MMRE	MBRE	PRED
Basic COCOMO	1.93	4.95	6.39	8.04	2.76	7.07	9.12	11.49	2.79	7.15	9.23	11.62
BAT	1.51	3.87	5.00	6.30	1.87	4.78	6.16	7.77	1.97	5.04	6.51	8.20
ACO	1.67	4.28	5.53	6.96	2.19	5.61	7.24	9.12	2.49	6.38	8.23	10.37
HACO-BA	1.06	2.72	3.51	4.42	1.35	3.47	4.47	5.63	1.61	4.12	5.31	6.69

scale data [18]. Deep learning frameworks are based on cutting-edge machine learning research and are used to create new features for Silicon Valley startups.

Machine learning is not the same as traditional programming. In traditional programming, the program we write instructs the computer on how to complete the task. Aside from that, in machine learning, we do not tell the computer exactly what to do. Alternatively, we provide training data, and the machine learning algorithm uses this training data to develop its own rules for completing the task. Deep neural networks are used in a variety of applications.

- (i) Image recognition, where you identify what objects are present in the images
- (ii) Image style transfer, where you can make a photograph look like, was painted in the style of a famous artist
- (iii) Language translation, where you translate from one human language to another
- (iv) Speech recognition, where you turn speech into text
- (v) Business problem, to solve a typical business problem estimating sales

However, deep learning faces many limitations, but not restricted to the lack of system programs to achieve optimal or ideal parameter values, manual configuration of deep learning architectures, and lack of standard training methods and algorithms. Therefore, researchers have proposed many methods including nature-inspired algorithms to alleviate the challenges.

The application of nature-inspired calculations in profound learning is limited due to the need for cooperative energy between profound learning and nature-inspired calculations. As a result of the lack of synergy between deep learning and nature-inspired algorithms, the application of nature-inspired algorithms in deep learning is limited [19]. In the context of big data analysis, the author demonstrated the role of nature-inspired algorithms in deep learning. The study, however, argued that nature-inspired algorithms have a very limited application in deep learning methods [18].

As the results are shown in Table 4, HACO-BA performs better among the other meta-heuristics algorithms. So, we applied the hybrid algorithmic approach to optimize the deep learning training process. HACO-BA is used to assign the best values of initial weights to the deep neural network. The proposed deep neural model is compared with [42, 46, 47] in terms of accuracy and time required for training. We have to find out the mean RE (MRE), mean magnitude of RE (MMRE), mean balanced residual error

(MBRE), and percentage of prediction (PRED) and then compared it to the NN model. The block diagram of the proposed systems is shown in Figure7.

*4.1. Data Acquisition and Processing.* In data acquisition and for further processing, three different data sets are used in this approach. NASA dataset is used which has complete data for software cost estimation for 93 different software projects. These data are occupied by distinctive NASA centers for a long time. NASA dataset along with COCOMO 81 and KEMERER dataset have various attributes. For these attributes, different parameters are selected for effort and time estimation. From which, 15 common variables along with their descriptions are listed in Table 5 are taken, and these attributes are input to the deep neural network at the input layer. As this is supervised machine learning, we have to provide a value as the result value in the output layer, which is the total amount of effort needed to build the software product.

The important step is that we need to preprocess our data. In order to train the deep neural network, we want to scale all the numbers in each column of our dataset to be between the value of 0 and 1. This is because if the numbers in one column are large but the numbers in another column are small, the neural network training will not work very well. One of the best ways to do this is to use the Min-MaxScaler object from the popular scikit-learn library. It is designed for exactly this purpose. In this method, first, we have to create a new MinMaxScaler, and then, we just need to pass in a feature range parameter, which tells it that we want all numbers scaled between 0 and 1.

*4.2. Model Building and Optimization.* In this section, model building and model optimization are presented.

*4.2.1. Model Building.* This step involves creating the model in TensorFlow, which is used to create and deploy supervised machine learning models. Supervised machine learning is a type of machine learning in which the model is trained by providing the data as input and the expected result for that data. It determines how to convert the input into the output. When developing and deploying a supervised machine learning model, we always adhere to a process known as the train, test, and evaluate flow.

First, we write the code for our machine learning algorithm. We will accomplish this by constructing a computational graph of operations, in which, to begin, we will

define each layer of the neural network and connect them so that data flow from the first to the last layer.

Then, we will add the placeholder node, which represents the data that will be fed into the neural network as input. Another placeholder node represents the neural network's output or the values predicted by the neural network. DNN has a total of five layers. There will be one input, one output, and three hidden layers between the neural network that will train to find the relationship between the inputs and the outputs. There are many different types of layers that can be used in a neural network, but we will stick with the most basic, a fully or completely connected neural network layer. That is, each node in each layer is linked to each node in the next layer.

Between layers, the first layer has 50 nodes, the second layer has 100 nodes, and the third layer has 50 nodes once more. Neurons are another name for nodes. Before training the model, the epoch is a hyperparameter to interpret in deep learning. When the entire dataset is passed forward and backward through the neural network only once, this is referred to as an epoch. We must set the training epochs to 100.

An epoch is another name for one full training pass over the training dataset. 50 epochs mean that we will do 50 iterations in our training loop to train our neural network, and similarly, 100 epochs mean 100 iterations.

Next, we need a way to measure the accuracy of the neural network's predictions. We will define the function that measures the each prediction accuracy during the training process. This is called a loss function. The loss function gets added to the graphs in its own operation. Then, we have to create an optimizer function that tells how we want to train the model.

When we run this function, it will perform one training step on our model. We will call this node the training operation. Bias is also an important parameter, and it means how far our forecast is from the actual value. In general, parameter algorithms have high biases, which makes them faster to learn and easier to understand, but they are usually less flexible.

The last part of defining this layer is multiplying the weights by the inputs and calling an activation function. An activation function outputs the result of the layer. We want the bias values for each node to default to zero.

*4.2.2. Model Optimization.* The deep neural network performance is optimized through training by examining the loss function results and balancing the weight of each neural network layer to produce better results by increasing the number of epochs. With deep neural networks, a lot of research have gone into the best initial values to use for weights. Weight is a very important part of the deep neural network if a set of given weights is not correct, it will take time to train the network and will not make correct predictions, so we have optimised it using an meta-heuristic algorithm. For this purpose, we have used grey wolf optimization (HACO-BA) algorithms which are a nature-inspired algorithm to give the best values for initial weights to train the network more precisely. We access a model's layer by using `model.layers`. Here, we set a layer's weights with `layer.setWeights()` to obtain from grey wolf algorithmic

optimizer. We have used code, like the following to set the optimised weights of each single layer: `model.layers[1].getWeight().setWeights(da.)`. Furthermore, we cannot set individual weights.

The variation in weights is decided by the learning rate. The learning rate is a parameter that appries the optimizer on how far to move the weights in the direction of the gradient. We have adjusted the learning rate of our model. By using HACO-BA for weight initializers, the proposed deep neural network model produces better results in less time as in contrast to the neural network. Results of the proposed procedure are shown is compared with wavelet neural network-firefly algorithm morlet activation function (WNN-FA-MORLET) [42], deep modified neural network (Deep-MNN) [46], and evolutionary cost-sensitive: deep belief network (ECS-DBN) [47] is also listed in Table 6.

*4.3. Performance Analysis and Evaluation of Deep Learning Model.* In this section, performance analysis and evaluation of the deep learning model is discussed.

*4.3.1. Performance Analysis.* The proposed deep neural network is compared with wavelet neural network-firefly algorithm morlet activation function (WNN-FA-MORLET) [42], deep modified neural network (Deep-MNN) [46], and evolutionary cost-sensitive deep belief network (ECS-DBN) [47] in terms of execution time required to train a model. Also, the proposed DNN is compared with the neural network in terms of accuracy achieved. Different software estimation datasets are given as input to the proposed DNN and find out whether the results change with a change in the dataset. In last, the proposed DNN is evaluated in terms of optimizing the evaluation matrices which is already defined in the subsection of Section 3 of this paper.

*4.3.2. Experimental Setup.* The model is built using the TensorFlow. TensorFlow is an open-source software or a library for building and deploying machine learning methods. The *Python* programming language and other different libraries are also used to build the models.

*4.3.3. Dataset.* As specified in the subsection of comparison of the existing meta-heuristic algorithms used for effort and cost estimation, three different most widely used dataset by the research community is used along with the China dataset which has a large number of software project data having 18 attributes in this experiments.

*4.3.4. Evaluation Matrices.* The difference between the start and end time of process execution of models is calculated which is the total time required to test and train the model. Also in order to measure the cost/effort, we will calculate the mean square error between what the neural network predicts and what we expect it to calculate. To do that, we will call the `tf.squared difference` function and pass in the actual prediction and the expected value. Also, our expected value is  $Y$ .

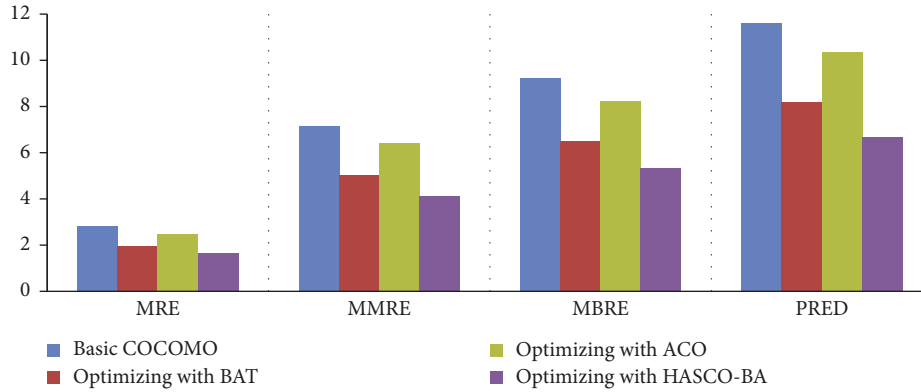


FIGURE 6: Comparison of KEMERER dataset.

TABLE 5: Selected parameters for effort and time estimation.

Variables	Description	Type	Role
Analyst's capability	Ability to learn and examine the system	Nominal	Input
Application experience	Basic application knowledge and skills	Nominal	Input
Process complexity	Event and tasks assessment that make the process	Nominal	Input
Database size	Large and complicated database	Nominal	Input
Modern programming practice	Updated method used for development	Nominal	Input
Programmer's capability	Knowledge and skill of programmer	Nominal	Input
Required software reliability	Failure-free probability of software	Nominal	Input
Schedule constraint	Earlier identify limitations on project schedule	Nominal	Input
Main memory constraint	Memory needs to effectively and efficiently completes several operations	Nominal	Input
Time constrain for CPU	Processing time to complete an action	Nominal	Input
Turnaround time	Amount of time required to complete a specific process	Nominal	Input
Virtual machine experience	Need for experience to operate on virtual systems	Nominal	Input
Use of software tools	Used of various modern framework	Flag	Input
Machine volatility	Experience and valuable knowledge to operate several machines	Nominal	Input
Effort	Efforts or resources required for development	Continuous	Output

Cost functions were included for a neural network, and the goal is to reduce the cost function. For this streamlined optimization problem, we use the GWO algorithm and variants of gradient descent where the model parameters (here weights and biases in the network) are rationalized in a way to reduce the cost function. All the data sets are used one by one for the training phase and for the testing phase. As this is a computational graph, there is no single start or end. We can start processing at any node in the graph, before we can perform any of the operations in our graph, we have to generate a session. Once the session object is created, we can ask it to run any operation in the graph. To train the model, we will call the training operation over and over. Each time the training operation runs, we will pass a new training data that will be used for that training pass. And then, we will check the current accuracy by calling the loss function. While the training process is running, we can watch the results graphically using a separate tool called Tensor Board. Different evaluation matrices which are specified in the sun section of Section 3 are also evaluated.

**4.4. Experimental Result.** Various experiments are carried out, and their results are compared in terms of time required for training, accuracy, and various evaluation matrices.

**4.4.1. Experimental 1.** In this experiment, the proposed scheme is differentiated with several DNN and NN models that include a deep modified neural network (Deep-MNN) [46], evolutionary cost-sensitive deep belief network (ECS-DBN) [47], and wavelet neural network-firefly algorithm morlet activation function (WNN-FA-MORLET) [42] in term of time required for training. With 50 epochs and with 100 epochs, we run the training and testing process, and the results reveal that the HACO-BA-DNN uses the less execution time appears in Figure 8 as compared to other nature-inspired algorithms.

**4.4.2. Experimental 2.** In this experiment, the proposed DNN is evaluated with neural network in terms of achieved accuracy. The process is executed upto 100 epochs, and the accuracy is accomplished by the proposed DNN approach is almost 98%. While NN achieved accuracy upto 85% on the same datasets as shown in Figure 9, the HACO-BA-DNN performs better in terms of accuracy as compared to NN.

**4.4.3. Experimental 3.** Various datasets of software estimation are given as input to the HACO-BA-DNN to find the change in results by changing the dataset. For this purpose,

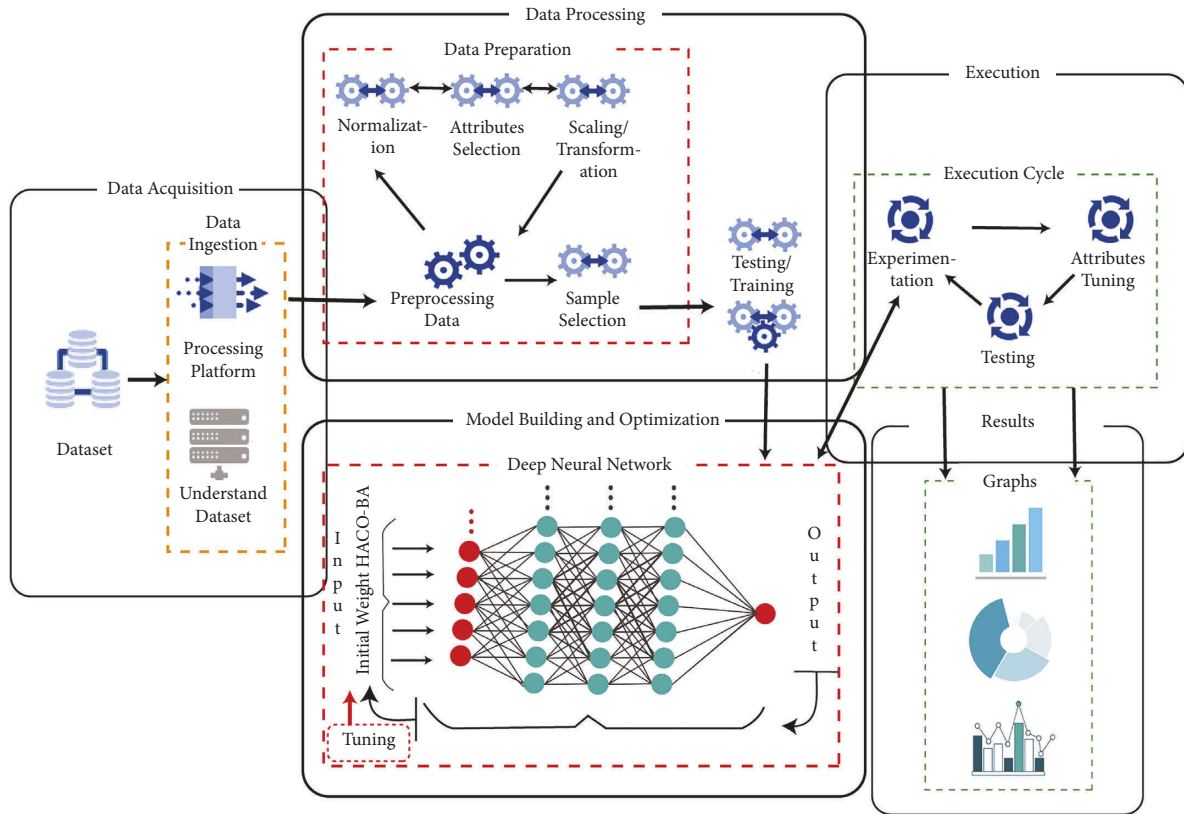


FIGURE 7: DNN model layers.

TABLE 6: Evaluation of execution time.

Methods	50 Epochs	100 Epochs
WNN-FA-MORLET	7.68	16.91
Deep-MNN	6.96	13.84
ECS-DBN	8.29	18.23
HACO-BA-DLL	5.32	11.7

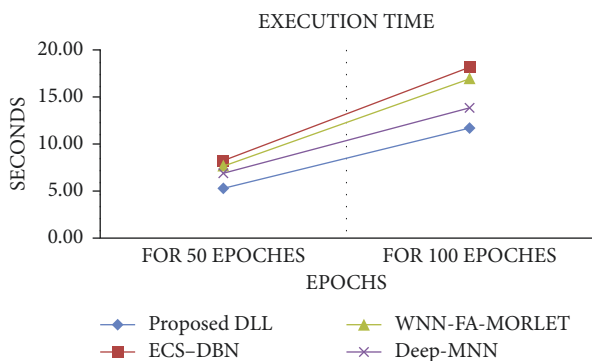


FIGURE 8: Comparison with literature.

we define the function which is known as the loss function that measures the accuracy of each prediction during the training process. Figure 10 demonstrates that there is no visible change seen in results when we change the dataset from NASA to COCOMO, KEMERER, or China.

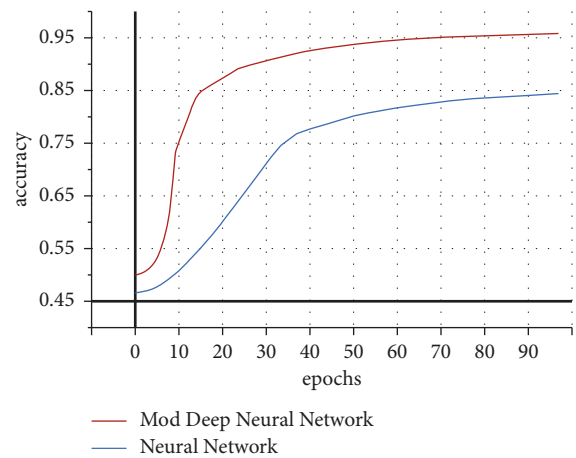


FIGURE 9: Comparison between NN and proposed DNN.

4.4.4. *Experimental 4.* The HACO-BA-DNN is evaluated as compared to NN by using various performance evaluation matrices, which are already defined in the subsection of comparison of the existing meta-heuristic algorithms. The results in Figure 11 show that the proposed DNN performs better in terms of reduction in MRE, MMRE, MBRE, and PRED. The smaller the value of the performance matrices shows that more results improve, and better software cost and effort estimation is achieved.

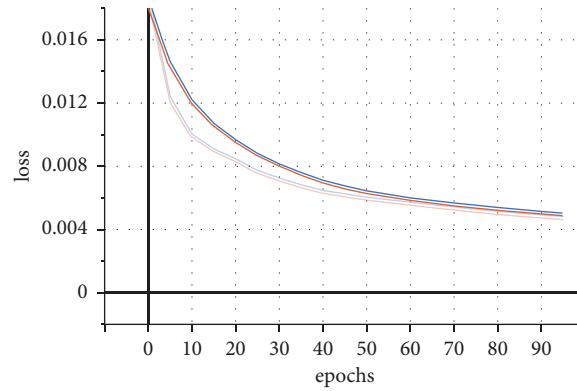


FIGURE 10: Using different datasets.

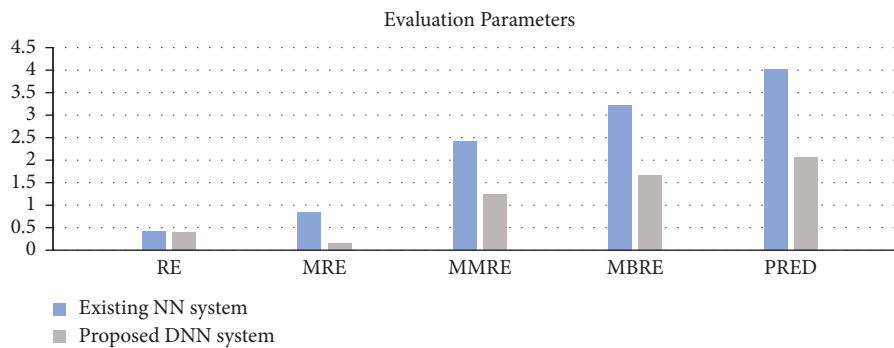


FIGURE 11: Comparison of NN with proposed DNN.

## 5. Discussions

5.1. *Answers to Research Questions.* RQ1, which of the meta-heuristic algorithm, has the lowest MRE, MMRE, MBRE, and PRED.

ACO and BAT along with their hybrid meta-heuristic algorithm, that is, HACO-BA has been implemented and their performance, is tested in terms of reduction in MRE, MMRE, MBRE, and PRED. As results show that all the algorithms reduced evaluation parameters as compared to the basic COCOMO parametric model, HACO-BA performs better among all other algorithms.

RQ2: whether the performance of the meta-heuristic algorithm changes/varies by changing the dataset.

All the meta-heuristic algorithms have been implemented, and their performance is tested and evaluated on three different publicly available data sets (NASA, COCOMO, and KEMERER), and we take MMRE as an evaluation parameter to check the performance of all three datasets. The results shows that the performance of HACO-BA is better as compared to the BAT, ACO, and Basic COCOMO as shown in Table 4.

RQ3: nature-inspired and meta-heuristic algorithms combined with deep learning can improve the software estimation process.

The proposed DLL takes less execution time as compared to other algorithms taken from the literature review. So it improves the software estimation process as shown in Figure 9.

RQ4: whether the performance of meta-heuristic-based deep learning algorithm changes by changing the dataset.

By changing the dataset, the performance of the proposed meta-heuristic deep learning architecture does not change. As shown in Figure 10, the two lines show that, with the passage of execution, both the data sets achieve almost the same accuracy.

RQ5: whether meta-heuristic-based deep learning algorithm performs better than the NN-based approach for software estimation.

Blue line shows the existing neural network approach.

Orange line shows the proposed deep neural network.

As shown in Figure 11, our proposed deep neural network performs better in terms of accuracy. NN takes more time/epochs as compared to HACO-BA-DNN to train its network to achieve better results in terms of software development effort reduction.

## 6. Conclusions

The proposed method investigates the efficacy of estimating efforts by combining ACO, BAT, and HACO-BA with COCOMO for effort estimation using optimised coefficients. To test the effectiveness of the proposed method, three datasets are used: Nasa, COCOMO 81, and KEMMER. MMRE values are improved in each optimised scenario, with HACO-BA outperforming all others. A new method that combines meta-heuristics and DNN is also introduced. As a

result of the results, it was determined that the optimised method produces better estimates than the basic method in terms of effort and cost estimation in all models. The experimental results show that the hybrid HACO-BA performs better for tuning COCOMO II than ACO and BA and that HACO-BA performs better in DNN optimization in terms of execution time and accuracy than NN.

## 7. Future Work

In the future, we will improve the estimation models by experimenting with new methods and incorporating cloud computing for estimating purposes in order to obtain more comprehensive results in the future. Researchers and practitioners in [59] and [61] used the Strawberry Plant heuristics approach for software cost estimation and for energy management. In [60, 62], Grey Wolf and Bacterial Foraging approaches are used in smart grids for energy management and heterogeneous generalized signcryption to maintain the data integrity for estimation.

## Data Availability

The data used to support the findings of this study can be obtained from the corresponding author.

## Conflicts of Interest

The authors state that they have no conflicts of interest.

## References

- [1] M. Usman, R. Britto, L. O. Damm, and J. Borstler, "Effort estimation in large-scale software development: an industrial case study," *Information and Software Technology*, vol. 99, pp. 21–40, 2018.
- [2] S. Mensah, J. Keung, M. F. Bosu, and K. E. Bennin, "Duplex output software effort estimation model with self-guided interpretation," *Information and Software Technology*, vol. 94, pp. 1–13, 2018.
- [3] N. Cerpa, M. Bardeen, C. A. Astudillo, and J. Verner, "Evaluating different families of prediction methods for estimating software project outcomes," *Journal of Systems and Software*, vol. 112, pp. 48–64, 2016.
- [4] V. Garousi, A. Coskuncay, A. Betin-Can, and O. Demirors, "A survey of software engineering practices in Turkey," *Journal of Systems and Software*, vol. 108, pp. 148–177, 2015.
- [5] I. Qasim, H. Tufail, A. Fatima, T. Rasool, and F. Azam, *Cost Estimation Techniques for Software Development: A Systematic Literature Review*, In *Proceedings of the International Conference on Electronics, Communications and Information Technology (ICECIT)*, pp. 38–42, 2018.
- [6] G. Gabrani and N. Saini, "Effort estimation models using evolutionary learning algorithms for software development," *Symposium on Colossal Data Analysis and Networking (CDAN)*, IEEE, vol. 2016, 2016.
- [7] N. Padhy, R. P. Singh, and S. C. Satapathy, "Software reusability metrics estimation: algorithms, models and optimization techniques," *Computers & Electrical Engineering*, vol. 69, pp. 653–668, 2018.
- [8] Y. Keim, M. Bhardwaj, S. Saroop, and A. Tandon, "Software cost estimation models and techniques: a survey," *International Journal of Engineering Research and Technology*, vol. 3, no. 2, pp. 1763–1768, 2014.
- [9] V. S. Dave and K. Dutta, "Neural network based models for software effort estimation: a review," *Artificial Intelligence Review*, vol. 42, no. 2, pp. 295–307, 2014.
- [10] S. Sarwar and M. Gupta, "Proposing effort estimation of cocomo ii through perceptron learning rule," *International Journal of Computer Application*, vol. 70, pp. 29–32, 2013.
- [11] S. Goyal and A. N. U. B. H. A. Parashar, "Machine learning application to improve COCOMO model using neural networks," *International Journal of Information Technology and Computer Science*, vol. 10, pp. 35–51, 2018.
- [12] P. Pospieszny, B. Czarnacka-Chrobot, and A. Kobylinski, "An effective approach for software project effort and duration estimation with machine learning algorithms," *Journal of Systems and Software*, vol. 137, pp. 184–196, 2018.
- [13] K. Zima, "The case-based reasoning model of cost estimation at the preliminary stage of a construction project," *Procedia Engineering*, vol. 122, pp. 57–64, 2015.
- [14] X. Huang, D. Ho, J. Ren, and L. F. Capretz, "Improving the COCOMO model using a neuro-fuzzy approach," *Applied Soft Computing*, vol. 7, pp. 29–40, 2007.
- [15] I. Kaur, G. S. Narula, R. Wason, V. Jain, and A. Baliyan, "Neuro fuzzy—COCOMO II model for software cost estimation," *International Journal of Information Technology*, vol. 10, pp. 181–187, 2018.
- [16] R. Malik, S. N. Gagandeep, W. Ritika, and J. Vishal, "Software Reliability Estimation Using COCOMO II and Neuro Fuzzy Method," *International Journal of Information Technology*, vol. 10, no. 3, 2018.
- [17] S. Shekhar and U. Kumar, "Review of various software cost estimation techniques," *International Journal of Computer Application*, vol. 141, no. 11, pp. 31–34, 2016.
- [18] F. Tahir and M. Adil, "An empirical analysis of cost estimation models on undergraduate projects using COCOMO II," in *Proceedings of the International Conference on Smart Computing and Electronic Enterprise (ICSCEE)*, IEEE, Shah Alam, Malaysia, July 2018.
- [19] R. Marco, "A systematic literature review on methods for software effort estimation," *Journal of Theoretical and Applied Information Technology*, vol. 97, no. 2, pp. 434–464, 2019.
- [20] R. Puri and I. Kaur, "Novel meta-heuristic algorithmic approach for software cost estimation," *International Journal of Innovations in Engineering and Technology (IJJET)* 5, vol. 2, pp. 456–463, 2015.
- [21] A. B. Nassif, M. Azzeh, L. F. Capretz, and D. Ho, "Neural network models for software development effort estimation: a comparative study," *Neural Computing & Applications*, vol. 27, no. 8, pp. 2369–2381, 2016.
- [22] S. Chalotra, S. K. Sehra, Y. S. Brar, and N. Kaur, "Tuning of cocomo model parameters by using bee colony optimization," *Indian Journal of Science and Technology*, vol. 8, p. 14, 2015.
- [23] R. K. Sachan, A. Nigam, A. Singh et al., "Optimizing basic COCOMO model using simplified genetic algorithm," *Procedia Computer Science*, vol. 89, pp. 492–498, 2016.
- [24] S. W. Ahmad and G. R. Bamnote, "Whale-crow optimization (WCO)-based optimal regression model for software cost estimation," *Sādhanā*, vol. 44, p. 94, 2019.
- [25] V. Venkataiah, R. Mohanty, J. S. Pahariya, and M. Nagaratna, "Application of ant colony optimization techniques to predict software cost estimation," in *Proceedings of the Computer Communication, Networking and Internet Security*, pp. 315–325, Springer, Singapore, May 2017.

- [26] N. Ghatasheh, H. Faris, I. Aljarah, and R. M. H. Al-Sayyed, "Optimizing software effort estimation models using firefly algorithm," *Journal of Software Engineering and Applications*, vol. 08, pp. 133–142, 2015.
- [27] S. M. S. Jafari and F. Ziaaddini, "Optimization of Software Cost Estimation Using harmony Search Algorithm," in *Proceedings of the Conference on Swarm Intelligence and Evolutionary Computation (CSIEC)*, IEEE, Bam, Iran, March 2016.
- [28] D. Shanthi, K. M. Rama, and G. Narsimha, "Application of partial swarm intelligence technique to predict software reliability," in *Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS)*, IEEE, Manhattan, NY, USA, 2017.
- [29] M. Padmaja and D. Haritha, "Software effort estimation using meta heuristic algorithm," *International Journal of Advanced Research in Computer Science*, vol. 8, pp. 121–125, 2017.
- [30] D. Nandal and O. Sangwan, "Software cost estimation by optimizing cocomo model using hybrid batgsa algorithm," *International Journal of Intelligent Engineering and Systems*, vol. 11, pp. 250–263, 2018.
- [31] F. S. Gharehchopogh, R. Rezaei, and B. Arasteh, "A new approach by using Tabu search and genetic algorithms in Software Cost estimation," in *Proceedings of the 9th International Conference on Application of Information and Communication Technologies (AICT)*, IEEE, Rostov on Don, Russia, October 2015.
- [32] R. Saljoughinejad and V. Khatibi, "A new optimized hybrid model based on COCOMO to increase the accuracy of software cost estimation," *Journal of Advances in Computer Engineering and Technology*, vol. 4, pp. 41–50, 2018.
- [33] Z. A. Dizaji, F. S. Gharehchopogh, and S. G. Farhad, "A hybrid of ant colony optimization and chaos optimization algorithms approach for software cost estimation," *Indian Journal of science and technology*, vol. 8, no. 2, p. 128, 2015.
- [34] F. S. Gharehchopogh, I. Maleki, and T. Akbar, "Using hybrid model of artificial bee colony and genetic algorithms in software cost estimation," in *Proceedings of the 9th International Conference on Application of Information and Communication Technologies (AICT)*, IEEE, Rostov on Don, Russia, October 2015.
- [35] M. Ahadi and J. Ahmad, "A new hybrid for software cost estimation using particle swarm optimization and differential evolution algorithms," *Informatics Engineering, an International Journal (IEIJ)*, vol. 4, p. 1, 2016.
- [36] A. Puspaningrum and R. Sarno, "A hybrid cuckoo optimization and harmony search algorithm for software cost estimation," *Procedia Computer Science*, vol. 124, pp. 461–469, 2017.
- [37] R. Delgado, Jairo, and R. Trujillo Rasúa, "Algoritmo meta-heurístico Firefly aplicado al pre-entrenamiento de redes neuronales artificiales," *Revista Cubana de Ciencias Informáticas*, vol. 12, pp. 14–27, 2018.
- [38] Z. Abdelali, H. Mustapha, and N. Abdelwahed, "Investigating the use of random forest in software effort estimation," *Procedia Computer Science*, vol. 148, pp. 343–352, 2019.
- [39] S. Fong, S. Deb, and X. S. Yang, "How meta-heuristic algorithms contribute to deep learning in the hype of big data analytics," in *Proceedings of the Progress in Intelligent Computing Techniques: Theory, Practice, and Applications*, pp. 3–25, Springer, Singapore, July 2018.
- [40] Z. Tian and S. Fong, "Survey of Meta-Heuristic Algorithms for Deep Learning Training," *Optimization Algorithms—Methods and Applications*, 2016.
- [41] P. Barmapalexis, A. Karagianni, G. Karasavvaides, and K. Kachrimanis, "Comparison of multi-linear regression, particle swarm optimization artificial neural networks and genetic programming in the development of mini-tablets," *International Journal of Pharmaceutics*, vol. 551, no. 1–2, pp. 166–176, 2018.
- [42] A. Kaushik and N. Singal, "A hybrid model of wavelet neural network and metaheuristic algorithm for software development effort estimation," *International Journal of Information Technology*, vol. 14, no. 3, pp. 1689–1698, 2022.
- [43] S. Quadri and Z. H. Wani, "An improved particle swarm optimisation-based functional link artificial neural network model for software cost estimation," *International Journal of Swarm Intelligence*, vol. 4, pp. 38–54, 2019.
- [44] E. Emary, H. M. Zawbaa, and C. Grosan, "Experienced gray wolf optimization through reinforcement learning and neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, pp. 681–694, 2018.
- [45] A. Kaushik, D. K. Tayal, K. Yadav, and A. Kaur, "Integrating firefly algorithm in artificial neural network models for accurate software cost predictions," *Journal of Software: Evolution and Process*, vol. 28, pp. 665–688, 2016.
- [46] M. K. Kodmelwar, S. D. Joshi, and V. Khanna, "A deep learning modified neural network used for efficient effort estimation," *Journal of Computational and Theoretical Nanoscience*, vol. 15, pp. 11–12, 2018.
- [47] S. Gu, R. Cheng, and Y. Jin, "Feature selection for high-dimensional classification using a competitive swarm optimizer," *Soft Computing*, vol. 22, pp. 811–822, 2018.
- [48] V. S. Desai and R. Mohanty, "ANN-cuckoo Optimization Technique to Predict Software Cost Estimation," in *Proceedings of the Conference on Information and Communication Technology (CICT)*, IEEE, Jabalpur, India, October 2018.
- [49] E. E. Miandoab and F. S. Gharehchopogh, "A novel hybrid algorithm for software cost estimation based on cuckoo optimization and k-nearest neighbors algorithms," *Engineering, Technology & Applied Science Research*, vol. 6, pp. 1018–1022, 2016.
- [50] A. Hussein, L. Chadad, N. Adalian, A. Chehab, I. H. Elhaji, and A. Kayssi, "Software-Defined Networking (SDN): the security review," *Journal of Cyber Security Technology*, vol. 4, pp. 1–66, 2020.
- [51] C. A. U. Hassan and M. S. Khan, "An Effective Nature Inspired Approach for the Estimation of Software Development Cost," in *Proceedings of the 16th International Conference on Emerging Technologies*, pp. 1–6, ICET, Islamabad, Pakistan, December 2021.
- [52] S. S. Ullah, I. Ullah, H. Khattak et al., "A lightweight identity-based signature scheme for mitigation of content poisoning attack in named data networking with internet of things," *IEEE Access*, vol. 8, pp. 98910–98928, 2020.
- [53] S. Hussain, I. Ullah, H. Khattak, M. A. Khan, C. M. Chen, and S. Kumari, "A lightweight and provable secure identity-based generalized proxy signcryption (IBGPS) scheme for Industrial Internet of Things (IIoT)," *Journal of Information Security and Applications*, vol. 58, no. 2021, Article ID 102625, 2021.
- [54] S. Hussain, I. Ullah, H. Khattak et al., "A lightweight and formally secure certificate based signcryption with proxy re-encryption (CBSRE) for Internet of Things enabled smart grid," *IEEE Access*, vol. 8, pp. 93230–93248, 2020.
- [55] S. S. Ullah, S. S. Ullah, S. Hussain, R. Alroobaea, and I. Ali, "Securing NDN-based Internet of health things through cost-effective signcryption scheme," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–13, 2021.



- [56] A. Dahou, M. Abd Elaziz, S. A. Chelloug et al., "Intrusion Detection System for IoT Based on Deep Learning and Modified Reptile Search Algorithm," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 6473507, 2022.
- [57] M. A. A. Al-qaness, A. A. Ewees, H. Fan, A. M. AlRassas, and M. Abd Elaziz, "Modified aquila optimizer for forecasting oil production," *Geo-Spatial Information Science*, pp. 1-17, 2022.
- [58] M. A. Al-qaness, A. A. Ewees, H. Fan, L. Abualigah, and M. A. Elaziz, "Boosted ANFIS model using augmented marine predator algorithm with mutation operators for wind power forecasting," *Applied Energy*, vol. 314, Article ID 118851, 2022.
- [59] M. S. Khan, A. U. H. Ch, A. S. Munam, and S. Azra, "Software Cost and Effort Estimation Using a New Optimization Algorithm Inspired by Strawberry Plant," in *Proceedings of the 24th International Conference on Automation and Computing (ICAC)*, IEEE, Newcastle Upon Tyne, UK, September 2018.
- [60] U. Hassan, C. H. Anwar, S. K. Muhammad et al., "Energy optimization in smart grid using grey wolf optimization algorithm and bacterial foraging algorithm," in *Proceedings of the International Conference on Intelligent Networking and Collaborative Systems*, Springer, Cham, August 2017.
- [61] M. S. Khan, C. H. A. U. Hassan, A. S. Hazrat, A. Ishtiaq, R. Asad, and J. Nadeem, "A new meta-heuristic optimization algorithm inspired from strawberry plant for demand side management in smart grid," in *Proceedings of the International Conference on Intelligent Networking and Collaborative Systems*, Springer, Cham, August 2017.
- [62] M. Rehman, H. Khattak, A. S. Alzahrani et al., "A lightweight nature heterogeneous generalized signcryption (hgsc) scheme for named data networking-enabled internet of things," *Wireless Communications and Mobile Computing*, pp. 1-20, 2020.

## Review Article

# A Review of the Role and Challenges of Big Data in Healthcare Informatics and Analytics

**Banan Jamil Awrahman , Chia Aziz Fatah, and Mzhda Yasin Hamaamin**

*Technical College of Applied Science, Sulaimani Polytechnic University, Sulaimani, Kurdistan Region, Iraq*

Correspondence should be addressed to Banan Jamil Awrahman; [banan.awrahman@spu.edu.iq](mailto:banan.awrahman@spu.edu.iq)

Received 12 May 2022; Revised 2 July 2022; Accepted 5 July 2022; Published 29 September 2022

Academic Editor: Mohamed Abdelaziz

Copyright © 2022 Banan Jamil Awrahman et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Healthcare has evolved with the development of technology to improve the quality of life and save lives. Today, big data is considered as one of the most essential and promising future technology areas and has been attracting the medical community's attention. As a result of big data, we can improve patient outcomes, personalize care, improve relationships between the patient and the provider, and decrease hospital costs. The effect of big data is very large since medical societies are known for their size, diversity of complexity, and a high degree of dynamism. Big data has been discussed from different viewpoints in recent years, protecting its involvement in many aspects, specifically those related to the healthcare system. Assembling health information, sharing data, and integrating health are essential in spreading health care. In addition, the security and privacy of data are critical since the data must be accessed from multiple locations within the distributed system. This paper review aims to understand the role of big data in healthcare issues aggregating data and the challenges associated with big data in healthcare. The papers that have been selected for review are from last year's research.

## 1. Introduction

Proper medical treatment for specific diseases will improve patient outcomes and decrease life-threatening conditions. It also reduces the side effects of drugs that impact their lives and medical waste products. Finding new drugs and equipment leads to further accuracy in the healthcare system [1]. Some sorts of medical equipment especially those which are continuously wearable record data and the high-velocity data requires fast processing; in a specific data source, the value may be limited, but in the public sector, it may get to a maximized value through fusion of electronic health records (EHRs) and electronic medical records (EMRs) [2]. CT scan for visualizing a patient's body, for instance, a patient's abdomen, is a plentiful source of high measurements data showing the abdomen with tiny details in such a high resolution that it is too beneficial in clinical settings and research for discovering abdominal features [3]. Web/mobile applications in health care have been expanded that enable patients to send their signs and symptoms to the

provider; those applications contain fundamental diseases, first aid, types of drugs, and also direct the patient to the specialist [4]. Health care system collects real-time biomedical signals (e.g., ECG, pulse oximetry, and blood pressure) in different places on mobiles, a healthcare application is installed, and health data are synchronized for analysis and storage by a cloud computing system [5] in health care; big data can be represented with the assistance of progressed information technology which observes information to make policy-making better; and a life chart can be used to research medical expenses and population aging, which applies evidence of policy-making [6]. Health care costs will be elevated with the aging population; Japan has begun using big data technologies for approaching and managing elderly persons, and big data analytics is used to attain information from complex and enormous datasets obtained from data mining [7].

This review provides a concise analysis of some productive efforts. In addition to the drawbacks and advantages of these technologies, privacy and security

TABLE 1: The scheme for diagnosis of diseases in a system.

Disease and illnesses	Diagnostic method bases	Health measurements
Diabetes mellitus	Scale-based, frequency-based	Blood glucose
Pulmonary disease	Scale-based, frequency-based	Oxygen saturation
Cardiac disease	Pattern matching, frequency	ECG
Infectious disease	Frequency-based, scale-based	Temperature
Hypotensive disease	Frequency-based, scale-based	Arterial pressure
Gastrointestinal tumors	Frequency-based	Video capsule pill

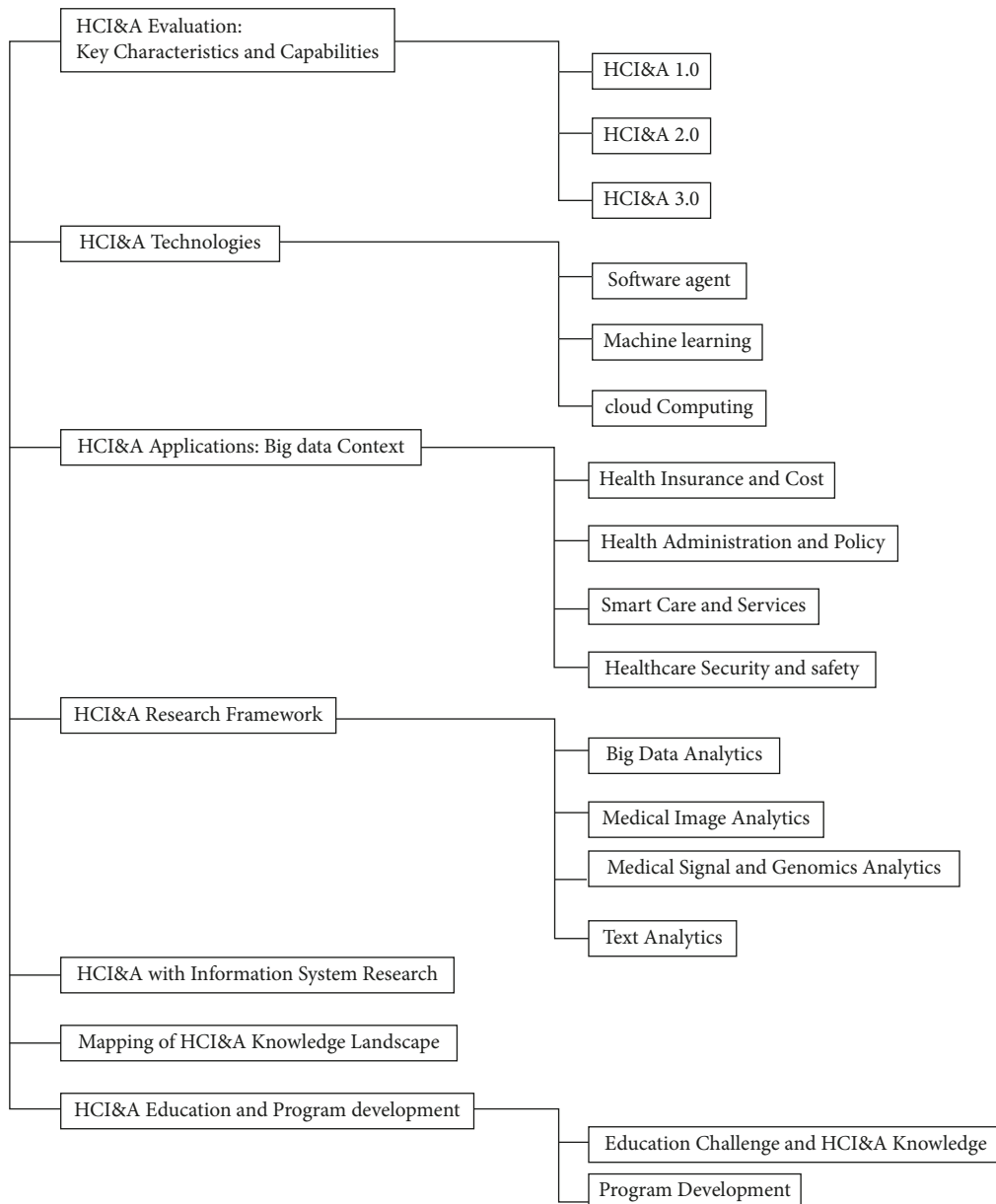


FIGURE 1: Overview of the healthcare informatics and analytics landscape: standards, technologies, applications, and emerging research.

have been discussed in phases of big data analytics in healthcare big data. Big data analytics has bridged the distinction between organized and unstructured data. The transition to an integrated data environment is a recognized hurdle to overcome. Big data's objective and

guiding concept is to gather more information and more insights from this information and has the capacity to forecast future occurrences. Several reputable healthcare firms expect a robust growth rate in the healthcare data sector.

## 2. Literature Review of Healthcare Data

Multiple forms of healthcare data include biomedical signals, genomic data, sensing data, biomedical images, and social media [8]. Genomic data analysis lets someone realize more about genetic markers, disease condition, consanguinity, and mutations; clinical text mining converts data from practical medical notes from disorganized format to applicable information, extraction of information, and natural language processing which extract helpful information from the massive volume of practical text. Social network analytics such as Web logs, Twitter, Facebook, social networking sites, and search engines helps to discover new health methods and worldwide health issues and trends based on different social media sources [9] before analyzing the severity of the disease; therefore, reasonable diagnostic patterns should be used. Table 1 represents the diagnostic plan for a definite diagnosis of the disease; the diagnosis is based on three conditions (frequency, pattern, and scale) [10]. Figure 1 identifies the importance of digitized health-related information we create; this figure contains seven layers showing personal health data. The privacy of individuals should be protected in this survey [11].

This is about wearable functions of the physiological sensors, and then referred to as mobile physiological sensor systems, designed for gathering user information through different sensors. These sensors measure a patient's vital signs, including ECG, temperature, oxygen saturation, pulse rate, and blood pressure. After that, those real-time data will be shown on the user's smartphone and sent to the health care cloud. Cloud systems can analyze and make classification using machine learning methods and store information in a private and secured manner. [12]. Healthcare data has been extended with a continuous stream of recent data elements and relationships, various data ranges from individual health information to epigenomes, and copious integration approaches are accepted, such as view integration (emerging and bringing together various databases), link integration (presentation in a web page), warehousing (setting data into a common database), and mash-ups (making a new web application from more than one web-based resource). All these methods make joining data flexible in many ways across sources but nevertheless consist of inadequate computable joined data or integration [13].

## 3. Healthcare Informatics and Analytics (HCI&A) Version 1.0

With the widespread adoption of database methods by different healthcare settings in the 1970s, HCI&A arose from the context of data management and analytics [13]. The goal of the Coral Gables Variety Children's Hospital's Patient Order Management and Communication System (POMCS) during that time was to accomplish three goals: raise income, increase employee productivity, and save money. [13]. These data management systems largely depend on technology for collecting, extracting, and analyzing health data. From a data-centric perspective, HCI&A may be compared to HCI&A 1.0, in which data is wholly organized,

homogeneous, and stored in relational database management systems (RDBMS). In addition, three other significant factors contributed to the medical domain's artificial intelligence and data analytics: the medical domain, the web, and data (see Figure 2). It consists primarily of Web 1.0 technologies, Health 1.0 apps, services, tools, and Medicine 1.0 solutions.

A hospital or healthcare institution distributes content on Web 1.0 without interacting with patients; it is primarily an online content repository. In the context of healthcare, Web 1.0 aims to create an online presence for healthcare providers that makes their information available at any time to all clients (primarily patients). In its cover of Web 1.0 tools and methods, HCI&A 1.0 encompasses the fundamentals of web technologies (HTML and HTTP), emerging web technologies (XML), and hypertext. Consumers and service providers cannot be involved in HCI&A 1.0 technologies. Provider-centric approaches are at the center of Medicine 1.0 and Health 1.0. Database technologies such as warehouses are used to integrate healthcare data management systems. Various statistical tools and data mining tools are also available in HCI&A 1.0 to classify, segment, cluster, and analyze health data. The leading commercial healthcare informatics systems from IBM, Oracle, and Microsoft already incorporate some HCI&A features. In addition to extracting, transforming, and loading data, we also have OLAP, database querying, data mining, and visualization packages within HCI&A 1.0. However, the software must also be able to perform some intelligence and analytical tasks.

## 4. Big Data Analytics in Healthcare System

Healthcare data digitization is the result of big data development and revolution. The rapid growth in data over the past few years led to the announcement of a new domain called big data. In information technology, the term "big data" is usually used to express enormous data that are too big and hard to deal with by the traditional database [14]. Intelligent healthcare systems, including big data analytics, make new and mobile health, saving medical costs and expanding efficiency [15]. Predicting pharmaceutical outcomes by predictive analytics, people who get the most benefit from pharmaceutical interventions are recognized, making pharmacists understand more about the side effects and risks of the medications [16]. Handling precision medicine is done by data collecting and management (sharing data, storing data, and privacy) to analytics (data merging, data processing, and visualization); compound and complex biomedical data which are enormous are becoming accessible due to biotechnologies progression, and analytics of big data is acquired to use these different data. It covers application sectors such as imaging, health, sensor, and bioinformatics [17]. For big data analytics, accuracy is essential; personal health records (PHRs) may contain typing errors, abbreviations, and mysterious notes; medical personal data input may contain errors, or it may be put in the wrong environment, which affects the efficacy of the collected data instead of getting uploaded by the professional

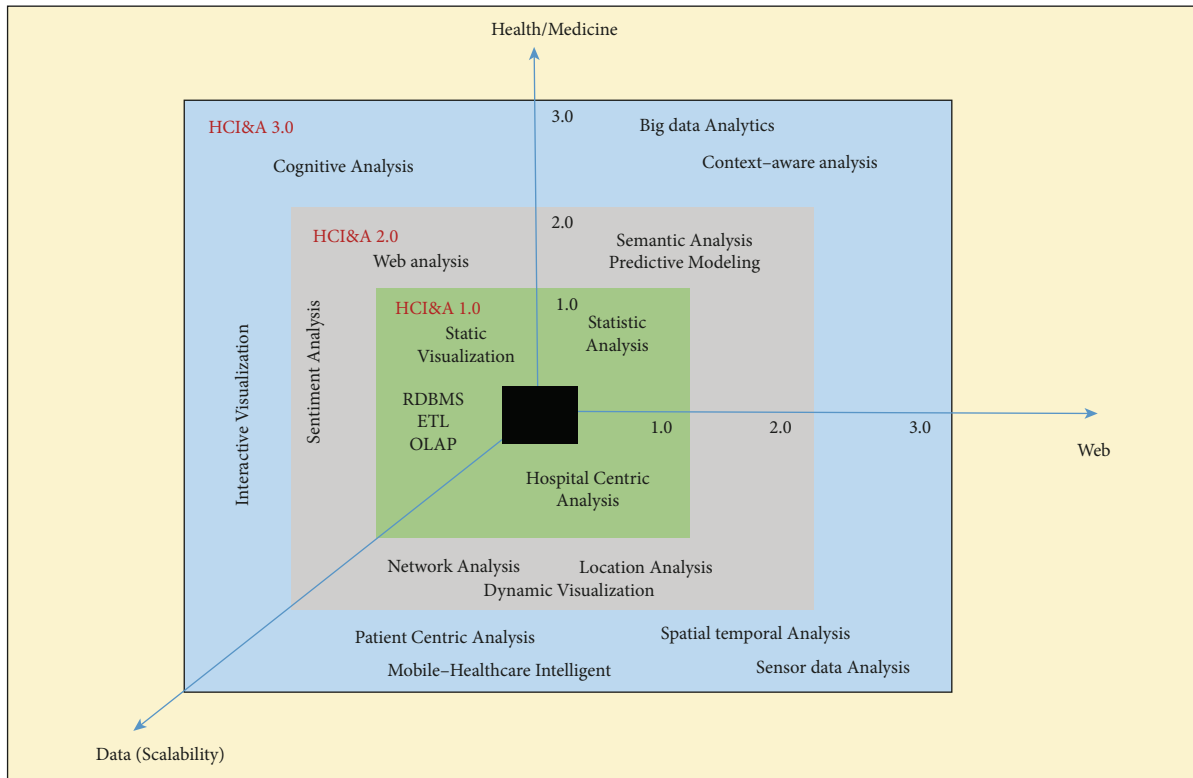


FIGURE 2: Analyses of health, web, and data in the context of health care.

trainee and medical practitioner in a clinical environment, and gathering data from social media may result in inaccurate prediction [18]. Fast-growing noise data is a significant problem; heterogeneous results are caused by various degrees of quality and completeness, which leads to false discoveries; there are two main problems, which are the inappropriate quality of data and biases because of absent randomization; big data value elevation are made by connecting various and analyzing all existing data [19]. Big data depending systems have progressed, including patient discharging records, electronic certificates of death, and medical claim data, which use the coding of International Classification of Diseases (ICD), and using big data courses in strategies of surveillance from the internet and social media has been preferred [20]. Data technologies like SQL databases have established healthcare processing. Some features like rational relationships and local access between logical and physical data spreading are significant to upgrading and performing parallel processing in database distribution [21].

Clinical and molecular information has been proposed in a big data-driven approach. Therapeutic medications and biomarkers are spotted in the approach. Following preclinical or clinical accuracy is accomplished by cross-species analysis; hence, the cost and time of biomarkers and therapeutics are decreased [22]. The primary function of the warehouse is for structured data and has a set of modules for unstructured data analysis. Initial accomplishments of substructures or frameworks were built for a big data paradigm. The framework used a Hadoop cluster for running modules, and

distributed counting ability is used in big data according to research [23]. Utilization of the enormous data storage and reliability of the Hadoop big data by the system makes a considerable reduction in storage and upgrade costs. Mobile applications are widespread, keeping doctors and patient users in touch, decreasing complex medical communications and increasing digitalization. Hadoop is a software framework that uses a master-slave. A group of essential background programs is mandatory to get Hadoop running in a completed cluster softly; it is also spread by a large amount of data and progress by Apache Foundation; it is likely to develop a distributed program is capable of dividing a large amount of program into small working units, making the cluster's ability to make high-speed storage [24]. MapReduce is based on rough set theory RST which is used for reducing attributes and includes these procedures for characteristic acquisition and accomplishes them on the MapReduce parallel large-scale rough set method which is used in runtime systems like the Phoenix, Twister, and Hadoop to get features from the big database by data mining two acceleration of computation of equivalence classes as done by using the framework structure of the (key, value) pair; MapReduce parallelizes traditional attribute reduction [25]. Tables 2 and 3 represent big data tools for health care [17].

Industry precised medicine is a sort of big data application in health issues, including the manufacturing of medical drugs and devices; it is considered as a strategic plan; this application benefits from IoT, industry, and multitopic. It has been suggested that it makes sense of big data with artificial intelligence, next-generation technology,

TABLE 2: Represent a list of several large companies which provide and supply services on big data analysis in the healthcare sector [1].

Data	Software	Description
Data integration or dataset or data source	Kafka, Sqoop	Tiny biosensors are placed on patients' body for collecting vital signs data in health applications, and the vital signs include blood pressure (BP), systolic and diastolic, respiratory rate (RR), heart rate (HR), oxygen saturation (spo2), and body temperature (BT)
Data decision and data storage	Apache Spark, Hadoop HDFS	That is responsible for storing data and processing it. That layer consists of two main tools, i.e., Hadoop and Apache Spark, also processing two data algorithms, patient archiving, emergency management, and clinical responses
Emergency detection and clinical response algorithm	Early warning score (EWS)	To verify abnormal situations
Patient classification and disease diagnosis	Machine learning algorithm	Machine learning tool and advanced analytics of huge datasets at high speed. Big data workspace tools are stored on Hadoop clusters for pattern insights discovered from massive data. Solutions to big data use cases by predictive analytics through platforms
Data retrieval and visualization	Hive, Spark SQL	Medical staff can access patients' records using the last platform for storing their data in HDFS and Hadoop. It comprises two data retrieval (spark SQL and hive) and just one graphing tool (Matplotlib). Obtaining data from the Hadoop storage system, which uses a set of criteria defined via queries, is based on the data retrieval tools. The retrieved data is usually stored in a file or displayed on a screen. Using graphics or plots statistically for data visualization is a graphical representation of the retrieved data; in our platform, each tool has been highlighted

TABLE 3: A scheme for some companies using big data.

Company name	Company place	Description	Company sits
Flatiron Health	New York, New York	Enough amounts of data points are utilized by Flatiron Health from cancer patients to develop research and obtain new patient care. This solution enhances all users, such as an oncologist, academic people, and researchers of life sciences connected to the treatment of cancer patients, as well as enables more learning from them	<a href="https://www.flatiron.com">https://www.flatiron.com</a>
Tempus	Chicago, Illinois	The most extensive library for clinical and molecular data in the world is Tempus which aims to provide more clinical context by medical professionals for cancer patients, and this platform is for collecting and organizing data from many aspects of oncology like pathology images, lab results, clinical notes, radiology scans, and oncology research acceleration and also assisting doctor specialists which helps them to make more informed treatment plans	<a href="https://builtinchicago.org">https://builtinchicago.org</a>
Pieces Technologies	Dallas, Texas	Collecting data from everything related to patients to make improvements in cost of care and quality is done through Pieces Technologies which is a cloud-based software company	
PeraHealth	Charlotte, North Carolina	For a patient's overall health, there is a universal scoring system that is a creator of the Rothman index; a peer-reviewed score collects the data in the electronic healthcare system, lab results, vital signs, and nurse assessment. A visual graph provides the score in real-time to recognize any changes and details also to avoid any complications for the patient	
Amitech	St. Louis, Missouri	To apply health data from modern data management to healthcare analytics, Amitech is used. It is specifically used to gather data for people health management solutions and collects physical health data in combination with behavioral health data to recognize risks and let the patients know their health	
SCIO Health	Hartford, Connecticut	For improving patient health, SCIO Health is used, which uses proprietary algorithms and integrated data for providing insights and solutions. The technology detects gaps in care that worsen health outcomes and cause more costs. Identification of these gaps assists medical professionals in detecting risky group patients and avoiding complications and insignificant hospitalization	
Hortonworks	Santa clara, California	Hortonworks are used for pharmaceutical data by pharmaceutical companies and researchers to obtain a better view. Companies can answer questions that were not possible previously because of billions of integrated records. This sparks much more effective research for clinical trials, faster marketing, better yields, and improved safety	

TABLE 4: HCI&amp;A used in various healthcare installations.

Empty cell	Health insurance and cost	Health administration and policy	Smart care and services	Healthcare security and safety
Data	Transactional documents payment album financial statement of the provider, user-generated content, and medical claims data	Official norms and regulations, information sources responses, and remarks from various organizations (i.e., doctors, nurses, patients, and other employees)	Computerized medical records (EHR), medical records patient comments and feedback, and molecular data DNA traces medical records (i.e., blood pressure, X-ray, and ECG)	Fraudulent records of data deviations, monetary data, geographic data, and social media records
Analytics	Detection of rare events, emotional evaluation, internet social network, study of statistical information integrating, segmenting, and clustering	Informational integration, administrative data, and ontological analytics textual examination performance, and appraisal rule of categorization and linkage	Mining associations and clusters, ontologies of health social media network, research data amalgamation monitoring and analysis of health, and network evaluation text analysis visualization	Linguistic text analytics, financial information analytics, GPS data evaluation sentiment analytics and social media network analytics, anomalous observation, criminal network investigation, and visualization
Applications	Funding and donation methodologies, recommendation methods, and system of transparent dispersion	Design of a resource, management policy, engagement and involvement of patients	Healthcare administration, support for healthcare decisions, healthcare service assessment, knowledge acquisition, patient vigilance, and universal healthcare	Criminal investigation, healthcare protection, patient safety, and intelligent care, recommendation system, and security administration
Contributions	To improve customer satisfaction, increase transparency and healthcare funding, and ensure responsibility	Enhanced administrative mobility, ensure appropriate actions at the proper time and location, remove congestion, and advocate for a strategy that is effective and efficient	Enhance healthcare (diagnosis, treatment, and therapy), patient-centered health care, develop an uninterrupted system for health monitoring	Enhance health care protection and reliability

and IoT [26]. Based on IoT technology, an intelligent healthcare framework has progressed for anyone during workouts; the Bayesian belief network uses an artificial neural network model to predict a patient's health-related susceptibility. There are four critical areas of big data analytics: model development, business models, data management, and visualization [10].

## 5. Challenges of Big Data in Healthcare Systems

Big data has been evolving, introducing challenges and problems caused by the exponential growth of healthcare data. The constant changes of big data present many challenges in analyzing, storing, and recovering huge amounts of data. Conventional or standard database systems cannot be used to process, store, and take information due to their massive and enormous volume [27]. Big data issues that generally happen in healthcare organizations are covered by four main categories [28, 29]: a huge amount of unstructured data are included in big clinical data like handwritten data and natural language, a reasonable degree of difficulty is brought by clinical big data's analysis, integration, and storage. It is insufficient for agencies to share structured data, and unstructured data sharing among organizations is more complicated. It is a great challenge how to effectively

mine an enormous amount of unstructured data. Big data has some characteristics. One of them is variability in data sources; medical data has potent timeliness; having appropriate moments of medical care is an example.

In the medical industry, data processing speed is in great demand particularly while patients' situation deteriorates quickly. The data privacy and security of the patients and ill persons are influenced by challenges and disputes with these real-time applications like cloud computing used to analyze data. Recently cloud computing has offered new possibilities for medical big data mining and sharing. Before cloud computing can become even more practical, several challenges must be overcome. [30]. First, cloud computing offers a simple and flexible way to mine resources. However, it elevates the risk of privacy disclosure. It is a fact that is clinically evident in clinical informatics. Second, importing or exporting an enormous amount of data in medicine to the cloud (petabyte). Network bandwidth increases the cost of data and restricts the speed [31] (see Table 4).

*5.1. Economic Challenges.* The medical field facilities of patients and health care providers such as doctors are dependent on paid services. It disproportionately negatively impacts technology advancements in connection with this process [32]. *Big Data Technology Challenges.* Being highly

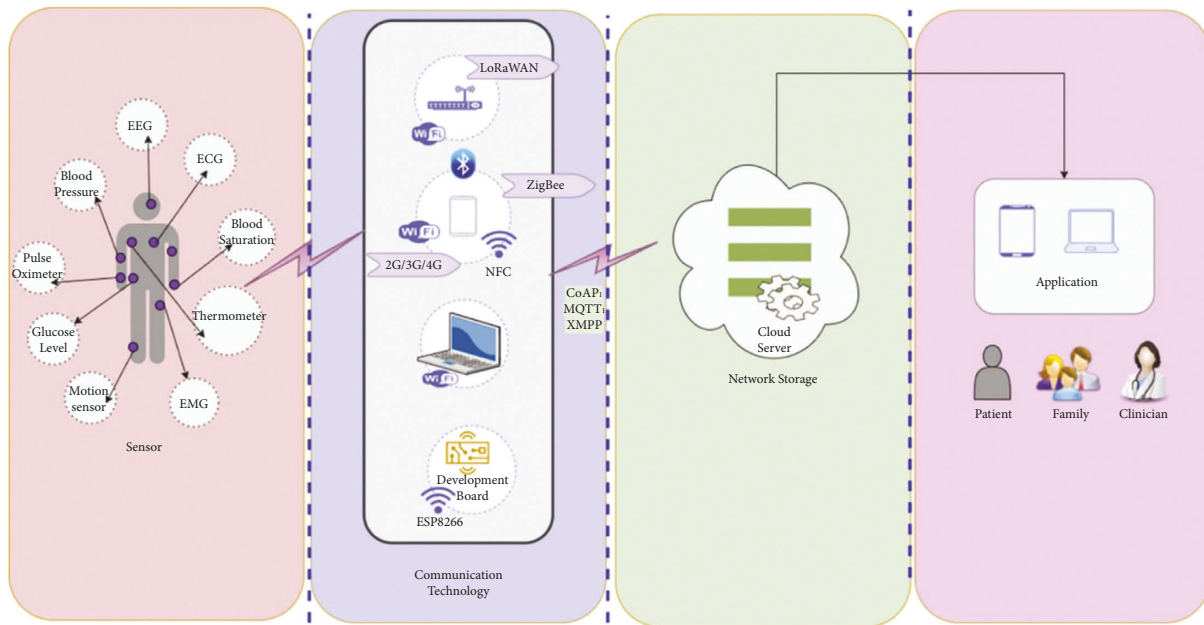


FIGURE 3: The communication technology utilized by the human body to transmit signals to the cloud.

fragmented and enormous, big data in health care leads to information quality and technology problems, making it a barrier to accomplishing healthcare vision [33]. Security and privacy issues along with the history of big data include the privacy of healthcare data which is serious because of potentially essential and sensitive information about individual healthcare providers. In order to make healthcare data unavailable in public, it must be secured from unauthorized access, preventing the data from attackers. This means security is the most important task, which is also a challenge [34].

**5.2. Privacy.** The most vital fault is the lack of intimacy and privacy. Big data must have access to almost everything, even social media life and private recordings, to have enough effects. However, because of revealing private information, the price is paid. Moreover, there is no patient freedom. However, there are regulations for stating medical recordings' privacy. However, they are not considered since it is believed that the information of someone should not be forbidden. At the same time, it is related to their health. The privacy risks associated with big data in health care have been stopped in articles such as big data privacy and security in health care [35].

**5.3. Health Information Systems on the Cloud.** The adoption of cloud-based platforms has improved and streamlined the design, development, and deployment of clinical information systems, hospital administrative information, and medical images [15]. Several such structures are in place to facilitate data collection (for example, the entities are often provided with mobile user interfaces to cloud services to gather and manage healthcare information). In addition,

these systems facilitate information exchange between various medical structures, hospitals, and patients since they integrate data in several ways. The performance of the system is rarely considered. Security and privacy, considered essential, are often at the center of their design (see Figure 3).

**5.4. Telepathology, Telehealth, and Disease Surveillance.** Telepathology services were envisioned as a possible outcome of combining robotic microscopy, video imaging, databases, and the then-new availability of broadband telecommunications in the 1980s. Many contributions demonstrating ICT applications have been presented, illustrating how ICTs can assist with telemedicine, telepathology, and disease monitoring. Research has been conducted on two problems: (1) general frameworks for most cases and (2) studies that focus on particular diseases, such as cancer detection, cardiovascular disorders, diabetes, Parkinson's disease, and Alzheimer's disease. These monitoring systems may then be utilized as a tool for large-scale research and as a means for customizing therapies (as in P4 Medicine). Likewise, surgery is expected to become more transparent in the future. Open surgery operating rooms often use video cameras for lighting. It allows an infinite number of viewers to view the surgical operation. Teleconsultation is possible with these instruments, eliminating the need for the consultant to be physically present. A remote consultant may use telepresence during surgery if an active camera holder is used and the remote consultant can move the camera. It is physically impossible for the surgeon to see the operating room when telesurgery is used. The availability of limited virtual pathways to fog services at the edge could assist in closing the gap when best-effort Internet connections are insufficient for some types of applications (e.g., to recreate the effect of a microscope locally). Providing



TABLE 5: Health conditions and their corresponding sensors and symptoms.

Diseases	Symptoms	Sensors
Stroke	In addition to losing your equilibrium, experiencing facial weakness, feeling numb on one side of your body, having difficulty speaking and interpreting, experiencing blurred vision in one or both of your eyes, and experiencing vertigo (vertigo caused by a severe headache)	Heart rate sensor, EEG, ECG, EMG, EOG, acceleration sensor, Samsung EDSAP, pulse oximeter, respiratory rate, blood pressure, and pulse oximeter
Lung cancer	The symptoms of chronic coughing include bloody coughing, bone pain, breathing problems, chest discomfort, headache, and weight loss	Sensors for measuring pressure, temperature, and acceleration, as well as FET-based biosensors
Blood cancer	Itchy skin, enlarged, painless lymph nodes in body parts, aching bones, fever, coughing, bone pain, and fever are symptoms of lymphoma	Electrochemical biosensors, optical biosensors, CMOS, thermometers, PPG, accelerometers, heart rate sensors, and blood flow sensors
Cardiac	Symptoms include chest discomfort, shortness of breath, weakness, and pain throughout the body	It has a heart rate monitor, pulse oximeter, accelerometer, glucose biosensor, blood pressure monitor, camera (image), microphone, acceleration sensor, PPG, pressure sensor, piezoelectric sensor, electrochemical sensor, and FET-based sensor.
Parkinson's	Symptoms of essential tremor include tremor, stiffness of muscles, slow movements, speech difficulties, mechanical movement difficulties, and difficulty writing	An accelerometer, a magnetometer, a gyroscope, an EMG, an EEG, and a bend sensor

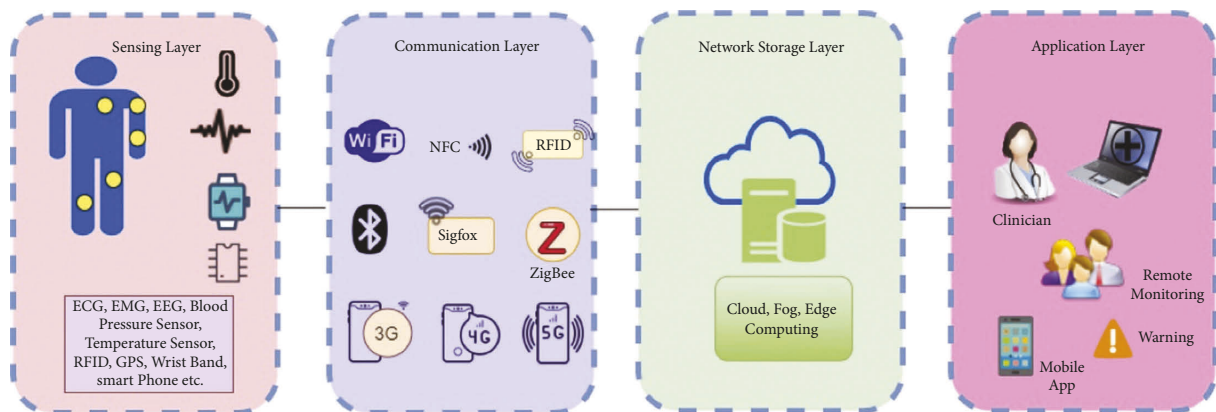


FIGURE 4: A four-layer system for IoT-based health monitoring.

remote federated sites with tools for offloading sophisticated image processing and data mining operations, it may, for instance, allow remote federated sites to cooperate on nontrivial diagnoses without experiencing increased cloud access latency (see Table 5).

## 6. Big Data Management in the Healthcare System

Healthcare activities generate large amounts of data. Analytical procedures should be used to derive actionable judgments from data management technologies. This section is divided into five subsections: machine learning-based, agent-based, cloud-based, heuristic-based, and hybrid-based. Further, in this section, the chosen articles are described in their approach, differences, advantages, and drawbacks (see Figure 4).

This section examines the most common machine learning techniques for managing extensive healthcare data along with their fundamental characteristics. In the last few

years, machine learning methods have been used to process large amounts of data based on artificial intelligence methods and historical databases. Therefore, machine learning techniques can be compellingly applied to this problem [31]. Machine learning algorithms play a significant role in managing massive biomedical data based on current issues in biomedical data [32].

## 7. Discussion on Intelligent Health Care

Sensor data are primarily unstructured in intelligent health care. Sensor-based health and wellness monitoring generate unstructured data beyond the human ability to process and analyze manually. There is a huge gap between the potential and the utility of such an enormous amount of unstructured data. The vast amount of unstructured data from streaming sensors is useless due to its variability and complexity. Data analytics pipelines for intelligent healthcare applications follow a similar process to the standard analytical method. Data management, processing, and finding are critically

important in health care [33, 34]. The correct data must be collected at the right time and context for an effective data discovery process. There needs to be an end to the division between numerous fields, such as medical science and computer science, for context-awareness in healthcare applications [35]. Data curation is, therefore, more useful when addressing effective data discovery when it comes to improving knowledge of patient physiological and psychological care.

**7.1. Interpretation of Data.** Predictive analytics may be more effective when combined with structured and unstructured EHR data. Clinical events can be extracted from EHR data, and comparable phrases can be categorized in semantic space. By concatenating their representation using semantic space, structured and unstructured data are integrated more efficiently than if the occurrences are represented separately. Using semantic spaces to extract clinical language from EHR, diverse and distributed representations can predict clinical outcomes effectively. The lack of an agreed-upon standard for terms, acronyms, and abbreviations further complicates the semantic categorization of datasets. This factor may impair the effectiveness of semantic classification based on similar terms. Various types of information can be collected from health records for purposes such as pharmacovigilance, phenotyping, and illness detection. Data from EHRs, EMRs, PHRs, and omics provide a wealth of information for many different medical fields. However, they should also be used to enhance healthcare. The model has been evaluated through interviews with domain experts, following the combination of clinical and genomic data for deep cancer phenotyping. In this study, real-time datasets could neither be used to assess the representation standard nor assess the suggested model. A robust knowledge base and accurate data modeling may facilitate using unstructured clinical notes from multiple institutions. The interpretation of data is equally important as obtaining usable information from various forms of health records, and this is called enhanced unstructured data analytics.

**7.2. Quality of Data.** The literature has identified that several quality parameters can be used to enhance and assess significant data quality, such as correctness, completeness, consistency, timeliness, objectivity, interpretability, and accessibility. Unstructured, heterogeneous, and noisy data add to the difficulty of this task because of their heterogeneity, lack of structure, noise, and the lack of a preset model. In addition to understanding psychological disorders, social media analytics helps to understand society's most prevalent illnesses. Social media analytics has most of the quality issues compared to other fields because postings, reviews, and comments cannot be standardized. Several linguistic issues impede clean analytics. It may be possible to increase analytical efficiency by using hashtags. However, computer, media, and healthcare knowledge are necessary to understand healthcare social media better. As part of effective healthcare analytics, database aggregation and data cleansing may reduce data heterogeneity, lack of structure, and other quality challenges.

## 8. Conclusion

This paper is a brief discussion of some successful work. Privacy and security have also been presented in phases of big data analytics along with the faults and benefits of these technologies in big healthcare data. Big data analytics has held the gap between structured and unstructured data. A well-known obstacle to overcome is the shift to an integrated data environment. The aim and principle of big data are gaining more information; more insights from this information and the ability to predict future data healthcare market show a rapid growth rate which several reliable healthcare companies project. However, in a short time, we have seen a range of analytics in use which has shown improvement effects on health care industry decisions. Computational experts have been forced by the exponential growth of medical data from different domains to design strategies to interpret and analyze various amounts of data. In every area, big data challenges are as follows: storing, searching, capturing, sharing, and analyzing data. Some extra challenges include real-time processing, data quality, privacy and security, and heterogeneous data. Also, healthcare data standards are among the challenges of big data analytics in healthcare systems. [36].

## Data Availability

It is a review paper and does not have any dataset.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] W. N. Price, *BIG DATA, PATENTS, AND THE FUTURE OF MEDICINE*, SSRN, New York, NY, USA, 2016.
- [2] Y. Zhang, M. Qiu, C. W. Tsai, M. M. Hassan, and A. Alamri, "Health-CPS: healthcare cyber-physical system Assisted by cloud and big data," *IEEE Systems Journal*, vol. 11, no. 1, pp. 88–95, 2015.
- [3] M. O. Ulfarsson, F. Palsson, J. Sigurdsson, and J. R. Sveinsson, "Classification of big data with application to imaging genetics," *Proceedings of the IEEE*, vol. 104, no. 11, pp. 2137–2154, 2016.
- [4] M. Panda, S. M. Ali, and S. K. Panda, "Big Data in Health Care: A Mobile Based Solution," in *Proceedings of the 2017 International Conference on Big Data Analytics and Computational Intelligence (ICBDAC)*, Chirala, Andhra Pradesh, India, March 2017.
- [5] L. A. Tawalbeh, R. Mehmood, E. Benkhelifa, and H. Song, "Mobile cloud computing model and big data analysis for healthcare applications," *IEEE Access*, vol. 4, pp. 6171–6180, 2016.
- [6] C. Wang, F. Li, L. Wang et al., "The impact of population aging on medical expenses: a big data study based on the life table," *BioScience Trends*, vol. 11, no. 6, pp. 619–631, 2017.
- [7] Y. Tsuji, "Medical privacy issues in ageing Japan," *Australian Journal of Adult Learning*, vol. 18, no. 1, 2017.
- [8] V.-D. Ta, C.-M. Liu, and G. W. Nkabinde, "Big data stream computing in healthcare real-time analytics," in *Proceedings of the 2016 IEEE International Conference on Cloud Computing*

- and Big Data Analysis (ICCCBDA), Chengdu, China, July 2016.
- [9] T. E. Al, "Big data stream computing in healthcare real-time analytics," *IEEE International Conference*, 2016.
  - [10] P. Verma and S. K. Sood, "Cloud-centric IoT based disease diagnosis healthcare framework," *Journal of Parallel and Distributed Computing*, vol. 116, pp. 27–38, 2018.
  - [11] D. Mendelson, "Legal protections for personal health information in the age of big data: a proposal for regulatory framework," *ETHICS, MEDICINE AND PUBLIC HEALTH*, vol. 3, no. 1, pp. 37–55, 2017.
  - [12] F.-Y. Leu, C.-Y. Ko, I. You, K. K. R. Choo, and C.-L. Ho, "A smartphone-based wearable sensors for monitoring real-time physiological data," *Computers & Electrical Engineering*, vol. 65, pp. 376–392, 2018.
  - [13] S. N. Murphy, P. Avillach, R. Bellazzi et al., "Combining clinical and genomics queries using i2b2 – three methods," *PLoS One*, vol. 12, no. 4, Article ID e0172187, 2017.
  - [14] R. Raja, I. Mukherjee, and B. K. Sarkar, "A Systematic Review of Healthcare Big Data," *Scientific Programming*, vol. 2020, pp. 1–15, Article ID 5471849, 2020.
  - [15] M. I. Pramanik, R. Y. Lau, H. Demirkan, and M. A. K. Azad, "Smart health: big data enabled health paradigm within smart cities," *Expert Systems with Applications*, vol. 87, pp. 370–383, 2017.
  - [16] I. Hernandez, Y. Zhang, and September, "Using predictive analytics and big data to optimize pharmaceutical outcomes," *American Journal of Health-System Pharmacy*, vol. 74, no. 18, pp. 1494–1500, 2017.
  - [17] P. Y. Wu, C. W. Cheng, C. D. Kaddi, J. Venugopalan, R. Hoffman, and M. D. Wang, "Omic and electronic health record big data analytics for precision medicine," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 2, pp. 263–273, 2017.
  - [18] J. Andreu-Perez, C. C. Y. Poon, R. D. Merrifield, S. T. C. Wong, and G. Z. Yang, "Big data for health," *IEEE J Biomed Health Inform*, vol. 19, no. 4, pp. 1193–1208, 2015.
  - [19] J. A. Sacristán and T. Dilla, "No big data without small data: learning health care systems begin and end with the individual patient," *Journal of Evaluation in Clinical Practice*, vol. 21, no. 6, pp. 1014–1017, 2015.
  - [20] L. Simonsen, J. R. Gog, D. Olson, and C. Viboud, "Infectious disease surveillance in the big data era: towards faster and locally relevant systems," *Journal of Infectious Diseases*, vol. 214, no. 4, pp. S380–S385, 2016.
  - [21] H. Salavati, T. J. Gandomani, and R. Sadeghi, "A robust software architecture based on distributed systems in big data healthcare," in *Proceedings of the International Conference on Advances in Computing, Communications and Informatics*, pp. 1701–1705, Udupi, India, September 2017.
  - [22] B. Wooden, N. Goossens, Y. Hoshida, and S. L. Friedman, "Using big data to discover diagnostics and therapeutics for gastrointestinal and liver diseases," *Gastroenterology*, vol. 152, no. 1, pp. 53–67.e3, 2017.
  - [23] S. Stephan and M. -R. Siadat, "Extensible query framework for unstructured medical data -- A big data approach," in *Proceedings of the 2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, pp. 455–462, Atlantic City, NJ, USA, November 2015.
  - [24] X. Zhang and Y. Wang, "Research on intelligent medical big data system based on Hadoop and blockchain," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, p. 7, 2021.
  - [25] J. Ni, Y. Chen, J. Sha, and M. Zhang, "Hadoop-Based Distributed Computing Algorithms for Healthcare and Clinic Data Processing," in *Proceedings of the 2015 Eighth International Conference on Internet Computing for Science and Engineering (ICICSE)*, pp. 188–193, IEEE, Harbin, China, November 2015.
  - [26] V. Özdemir and N. Hekim, "Birth of industry 5.0: making sense of big data with artificial intelligence, "the internet of things" and next- generation technology policy," *OMICS: A Journal of Integrative Biology*, vol. 22, no. 1, pp. 65–76, 2018.
  - [27] A. Rishika Reddy and P. Suresh Kumar, "Predictive big data analytics in healthcare," in *Proceedings of the Conference: 2016 Second International Conference on Computational Intelligence & Communication Technology (CICT)*, Ghaziabad, India, February 2016.
  - [28] D. V. Dimitrov, "Medical internet of things and big data in healthcare," *Healthcare Informatics Research*, vol. 22, no. 3, p. 156, 2016.
  - [29] B. K. Sarkar, "Big Data for Secure Healthcare System: A Conceptual Design," *Complex Intell. Syst*, vol. 3, no. 2, pp. 133–151, 2017.
  - [30] J. Chen, F. Qian, W. Yan, and B. Shen, "Translational biomedical informatics in the cloud: present and future," *BioMed Research International*, vol. 2013, p. 1, 2013.
  - [31] L. Hong, M. Luo, R. Wang, P. Lu, W. Lu, and L. Lu, "Big Data in Health Care: Applications and Challenges," *Data and Information Management*, vol. 0, no. 0, 2019.
  - [32] L. Wang and C. A. Alexander, "Big data analytics in healthcare systems," *International Journal of Mathematical, Engineering and Management Sciences*, vol. 4, no. 1, pp. 17–26, 2019.
  - [33] F. Firouzi, B. Farahani, M. Ibrahim, and K. Chakrabarty, "Keynote Paper: From EDA to IoT eHealth: Promises, Challenges, and Solutions," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 37, no. 12, pp. 2965–2978, 2018.
  - [34] A. G. Alexandru, I. M. Radu, M. L. Bizon, and June, "Big data in healthcare - opportunities and challenges," *Informatica Economica*, vol. 22, no. 2/2018, pp. 43–54, 2018.
  - [35] K. Abouelmehdi, A. Beni-Hessane, and H. Khaloufi, "Big healthcare data: preserving security and privacy," *J Big Data*, vol. 5, no. 1, 2018.
  - [36] S. Dash, S. K. Shakyawar, M. Sharma, and S. Kaushik, "Big data in healthcare: management, analysis and future prospects," *Journal of Big Data*, vol. 6, no. 1, 2019.

## Research Article

# An Improved Image Classification Method for Cervical Precancerous Lesions Based on ShuffleNet

Shan Fang <sup>1</sup>, Jiahui Yang,<sup>1</sup> Minghui Wang,<sup>1</sup> Chunhui Liu <sup>2</sup>, and Shuang Liu <sup>1</sup>

<sup>1</sup>College of Quality and Technical Supervision, Hebei University, Baoding 071002, China

<sup>2</sup>Affiliated Hospital of Hebei University, Baoding 071002, China

Correspondence should be addressed to Chunhui Liu; liuchs@hbu.edu.cn and Shuang Liu; whlius@hbu.edu.cn

Received 27 May 2022; Revised 8 July 2022; Accepted 16 July 2022; Published 13 September 2022

Academic Editor: Mohamed Abdelaziz

Copyright © 2022 Shan Fang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the rapid development of deep learning, automatic lesion detection is used widely in clinical screening. To solve the problem that existing deep learning-based cervical precancerous lesion detection algorithms cannot meet high classification accuracy and fast running speed at the same time, a ShuffleNet-based cervical precancerous lesion classification method is proposed. By adding channel attention to the ShuffleNet, the network performance is improved. In this study, the image dataset is classified into five categories: normal, cervical cancer, LSIL (CIN1), HSIL (CIN2/CIN3), and cervical neoplasm. The colposcopy images are expanded to solve the problems of the lack of colposcopy images and the uneven distribution of images from each category. For the test dataset, the accuracy of the proposed CNN models is 81.23% and 81.38%. Our classifier achieved an AUC score of 0.99. The experimental results show that the colposcopy image classification network based on artificial intelligence has good performance in classification accuracy and model size, and it has high clinical applicability.

## 1. Introduction

Cervical cancer is the fourth most common female cancer. The statistics of WHO show roughly 604,000 new cases worldwide in 2020, accounting for 6.5% of all new cancer cases in women [1]. The early cure rate of cervical cancer is high, but the lack of signs and symptoms at this stage hinders the early diagnosis. A successful screening program can avoid cervical cancer death and reduce the incidence and persistence of the disease [2]. According to statistics, more than 311,000 cervical cancer deaths occur every year. Due to the lack of experienced health care staff and insufficient funds for the screening system, cervical cancer screening facilities are very scarce in developing countries [3]. Therefore, it is necessary to use automated and effective screening methods, to reduce the cost of early detection of cervical cancer. Cervical cancer screening follows the following workflow: HPV test, cytology or PAP smear test, colposcopy, and biopsy [4]. The PAP smear image screening is to take a small number of cell samples from the cervix of the uterus, placing them on glass slides, and then study

whether they are abnormal under a microscope. This method is time-consuming and depends on the experience of pathologists. Different pathologists will see different results in the same film. The HPV test is a DNA test. PAP smear and HPV test are very expensive treatments with low sensitivity. Therefore, colposcopy is widely used in developing countries. Colposcopy identifies cervical lesions by using a low magnification microscope under a strong light source [5]. Its accuracy highly depends on the skills of physicians. There are significant differences in the detection rate of lesions among different colposcopy physicians. This has aroused people's attention to the insufficient diagnosis of lesions (including missed diagnosis of cervical cancer) and excessive diagnosis of lesions [6–9]. Excessive diagnosis of lesions may lead to the excessive treatment of low-grade cervical lesions, increasing the risk of infection and economic burden [10].

In recent years, deep learning has gradually become popular in the field of medicine. The purpose of medical image processing is to restore the original unclear image, to highlight some characteristic information in the image, or to

classify the image. Medical images include MRI, CT, ultrasound images, and blood smear images [11, 12]. Convolutional neural network (CNN) is an important end-to-end deep learning model [13], which is mainly used in image recognition, segmentation, and target detection in medical image processing. Ai-assisted colposcopy can help colposcopy specialists improve their diagnostic performance, optimize clinical workflow, and relieve pressure on colposcopy physicians and hospitals, which has great potential to improve cervical cancer screening performance.

We propose a method for the classification of cervical precancerous lesions based on deep learning. The main contributions of this paper are as follows:

- (i) Different grades of cervical precancerous lesions, cervical neoplasm, and cervical cancer were classified.
- (ii) A deep inverted residual network based on the improved additional channel attention of ShuffleNet is proposed.
- (iii) Compared with the traditional residual network, the inverted residual network can not only ensure the automatic extraction of features in the image but also reduce the amount of calculation and improve the calculation speed of the model.

The structure of this paper is as follows: Section 2 introduces the proposed deep learning model. Section 3 describes data sources and processing. Section 4 is experiment and analysis. Section 5 concludes this work.

## 2. Materials and Methods

**2.1. Depthwise Separable Convolution.** Deep CNN networks such as ResNet [14] and DenseNet [15] have greatly improved the accuracy of image classification. However, in addition to accuracy, computational complexity is also an important index to be considered by the CNN network. Complex networks may run slowly. Some specific scenes, such as an unmanned vehicle, need low latency, and edge computing devices also need small models that are both accurate and fast. To meet this demand, lightweight deep learning networks such as MobileNet [16] and ShuffleNet [17] have been proposed, which achieve a good balance between speed and accuracy.

To speed up the calculation speed of the network and reduce the amount of calculation, MobileNet proposes depthwise separable convolution. For the traditional convolution, an input feature graph with a size of  $(W, H, C_{in})$  is used to obtain an output feature graph with a size of  $(W, H, C_{out})$  through convolution operation using an  $N \times N$  convolution kernel. At this point, the computational quantity is

$$W \times H \times C_{in} \times C_{out} \times N \times N. \quad (1)$$

Depthwise separable convolutions are divided into depthwise convolutions and pointwise convolutions. Depthwise convolutions are equivalent to using the convolution kernel with the number of channels of 1 to perform separate convolution operations on each channel of the input feature

map. The feature map with the same number of output and input channels needs to be multiplied  $W \times H \times C_{in} \times N \times N$  times. Pointwise convolution, a simple  $1 \times 1$  convolution, needs  $W \times H \times C_{in} \times C_{out}$  times of multiplication calculation. Compared to ordinary convolution, the calculation amount of depthwise separable convolution can be reduced:

$$\frac{W \times H \times C_{in} \times N \times N + W \times H \times C_{in} \times C_{out}}{W \times H \times C_{in} \times C_{out} \times N \times N} = \frac{1}{C_{out}} + \frac{1}{N \times N}. \quad (2)$$

**2.2. Inverted Residual Network with Additional Channel Attention.** ShuffleNet has similar ideas with MobileNet, Xception [18], and ResNet. It uses channel shuffle and depthwise separable convolution to optimize the residual structure of ResNet, which not only ensures the network accuracy but also improves the operational efficiency of the model. Unlike the traditional residual module, which directly integrates the features of the deep networks and nondeep networks obtained through multiple convolutions, the inverted residual module divides the input feature map into two batches X1 and X2, X2 through depthwise separable convolution and twice  $1 \times 1$  convolution + batch standardization + activation function, X1 and X2 are fused with deep and nondeep features, and finally, channel shuffle is used to mix deep and nondeep features. Suppose that the input layer is divided into  $G$  groups, and the total number of channels is  $G \times n$ . First, divide the channel into two dimensions  $(G, n)$ , then transpose these two dimensions into  $(n, G)$ , and finally reshape them into one dimension  $G \times n$ . The ShuffleNet structure model is shown in Figure 1. The channel shuffle process is shown in Figure 2.

To make the classification more accurate, we add the Squeeze-and-Excitation Networks (SE) [19] and the Selective Kernel Networks (SK) [20] to the model, respectively. The SE model is shown in Figure 3. Firstly, a feature map  $U$  with a total number of channels  $C$  and a size of  $H \times W$  is flattened into a feature vector of  $(1, 1, C)$  by a global pooled Fsq shown as follows:

$$Z_c = F_{sq}(U_c) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W U_c(I, j). \quad (3)$$

The activation function and linear mapping are added to the feature vector to add more nonlinear conditions, which can better fit the complex correlation between channels. Finally, the calculated channel features are multiplied by the original feature map to obtain the output of channel attention. The SE model strengthens the important features and weakens the unimportant features by controlling the size of the channel proportion, to make the extracted features more directional.

Channel attention is allowed to be inserted between each feature map. After the SE channel attention is inserted into the depth-separable convolution, feature extraction of channel dimension is carried out on the depthwise separable convolution output. The inverted residual network structure model fusing the SE module is shown in Figure 4.

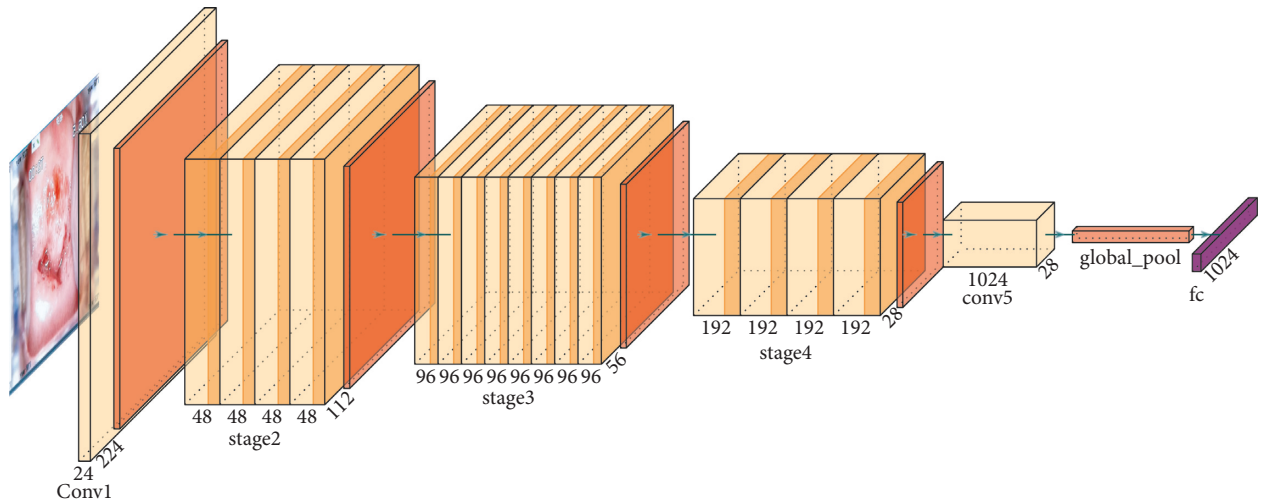


FIGURE 1: ShuffleNet structure model.

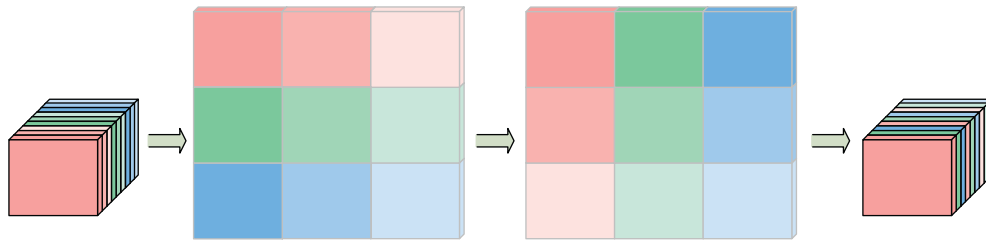


FIGURE 2: Channel shuffle process.

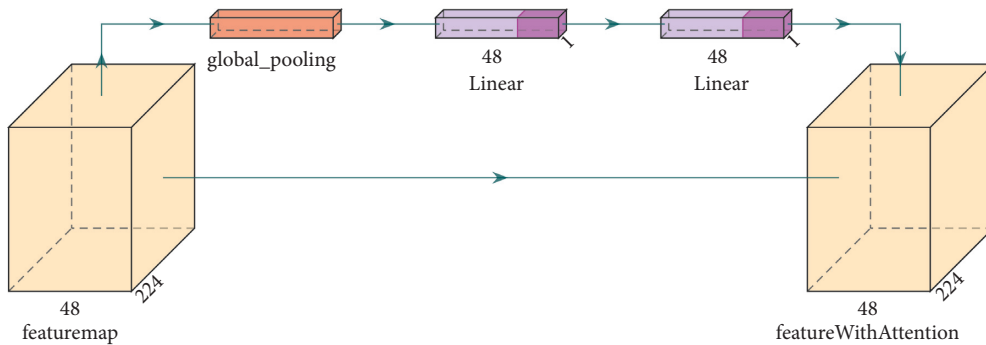


FIGURE 3: Squeeze-and-Excitation networks structure model.

SK is mainly the same as SE. The difference is that SENet performs attention on the channel, while SKNet performs attention on the convolution kernel. SKNet uses convolution check feature maps of different sizes in the network to extract features of different scales and then extracts channel attention after fusion of features of different scales. Its network model is shown in Figure 5.

To compare with SE, SK is also used in depthwise separable convolution. The inverted residual network structure model fusing the SK module is shown in Figure 6.

First, the input feature maps are computed by a depthwise separable convolution conv\_1 with a convolution kernel size of  $3 \times 3$ , and a depthwise separable convolution with a convolution kernel size of  $3 \times 3$  and a dilation factor of 2 in different scales; then the two output feature maps are summed for global pooling, and the pooling layer is computed similar to the SE channel attention; subsequently, the output of two-channel features are multiplied with conv\_1 and conv\_2 in the channel dimension to obtain two feature maps of mixed channel

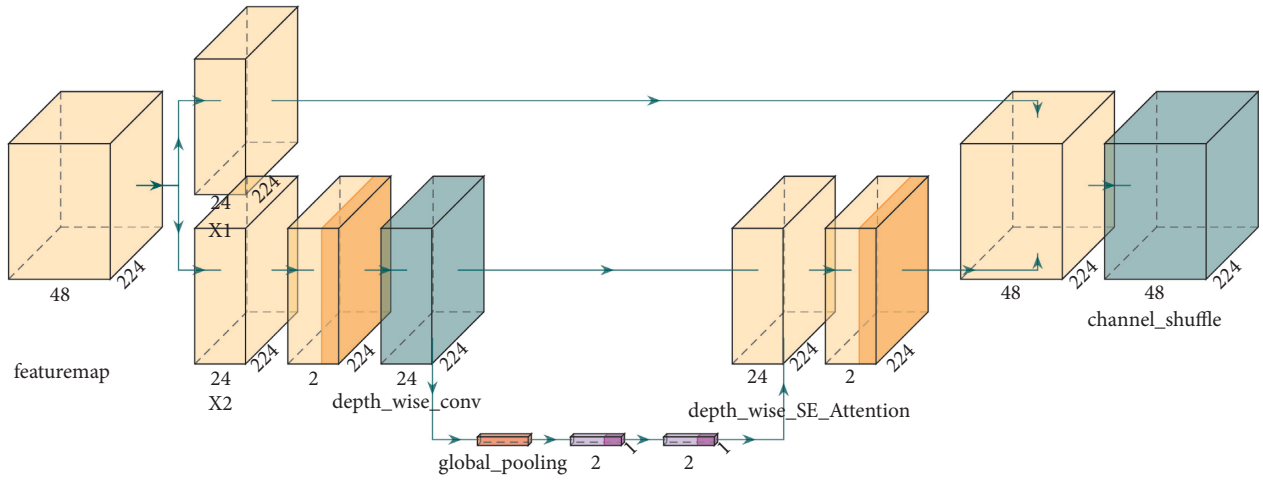


FIGURE 4: Inverted-residual-SE structure model.

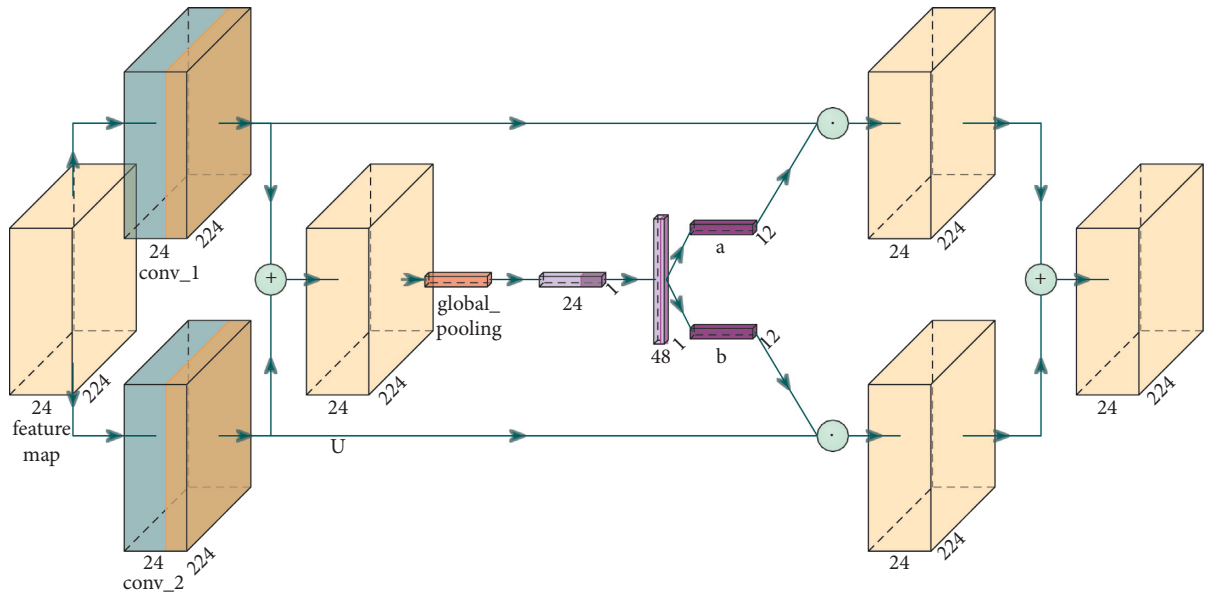


FIGURE 5: Selective Kernel attention structure model.

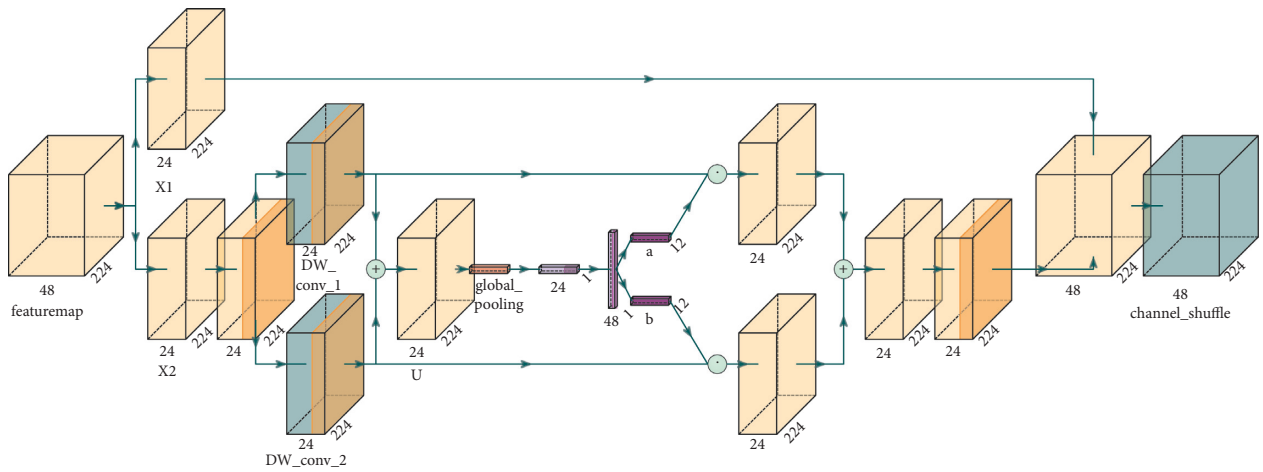


FIGURE 6: Inverted-residual-SK structure model.

attention at different scales, and then the two feature maps are summed to obtain the SK attention output features.

### 3. Data Source and Processing

*3.1. Cervigram Dataset.* The cervical cancer screening dataset was provided by the Department of Gynecology, Affiliated Hospital of Hebei University (as show in Table 1). The dataset consisted of colposcopy images of different grades of precancerous lesions (normal, CIN1, and CIN2/3), cervical neoplasm, and cervical cancer. There are 1,189 patients, totaling 6,996 images.

*3.2. Dataset Making Principle.* In this study, data split into training and validation subsets using a 90% to 10% ratio.

Since the uneven distribution of the provided dataset in each category and the number of samples is small, data augmentation is used to add images for five categories (normal, cervical cancer, HSIL, LSIL, and cervical neoplasm). Data augmentation is used to improve the overall structural security of the trained model. There are two ways to enhance the data: one is to get new images; another method is to augment the data, i.e., to create more available data using already available data such as flips, translations, or rotations to make the neural network more generalizable. Three data augmentation methods used in this paper are as follows:

- (i) Randomly cropping the image size to  $224 \times 224$
- (ii) Image standardization processing
- (iii) Random horizontal and vertical image flipping

### 4. Experiments and Performance Analysis

*4.1. Experimental Conditions.* To ensure the iterative efficiency and improve the model stability and generalization ability, in this study, the network parameters are optimized by stochastic gradient descent (SGD) algorithm using nesterov gradient descent with weight decay of  $1e-4$ , learning momentum of 0.9, and several single batch treatments of 32. Each model is trained for 100 cycles, and the initial learning rate is set to 0.05. The CNN algorithm is implemented in PyTorch coding framework. Model training and evaluation are conducted using Intel (R) Xeon (R) Gold 6240 CPU@ 2.60 GHz and NVIDIA RTX 2080ti GPU. All programs run on Ubuntu 18.04.5 LTS.

*4.2. Evaluation Metrics.* To evaluate the algorithm effectively, this paper uses training loss and model accuracy for measurement in the training phase. In the test phase, this paper introduces the confusion matrix as the basic evaluation criterion, and the confusion matrix contains four parts of information:

- (i) TN, which is the true negative, can represent the number of negative samples predicted as negative

TABLE 1: Cervix types in cervigram dataset.

Type	Label	Number of images
Normal	0	2352
LSIL	1	780
HSIL	2	2532
Cervical cancer	3	408
Cervical neoplasm	4	924

- (ii) TP, which is the true positive, can represent the number of positive samples predicted as positive
- (iii) FN, which is the false negative, can represent the number of positive samples predicted as negative
- (iv) FP, which is the false positive, can represent the number of negative samples predicted as positive

Since the proposed model is a multiclassification model, accuracy, precision, recall, and F1-scores can be calculated according to the above four indicators. The area-under-the-curve (AUC) score and the confusion matrix are also used to evaluate the performance of the model. The classification accuracy, precision, recall, and F1 score can be obtained by (4)–(7).

$$\text{Accuracy (\%)} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \times 100, \quad (4)$$

$$\text{Precision (\%)} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100, \quad (5)$$

$$\text{Recall (\%)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100, \quad (6)$$

$$\text{F1 - score (\%)} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \times 100. \quad (7)$$

The receiver operating characteristic (ROC) curve is a comprehensive index that shows continuous changes in sensitivity and specificity. According to the position of the curve, the whole graph is divided into two parts. The area under the curve is called AUC. The higher the AUC score, the better the performance of the classification model. The confusion matrix reflects the confusion caused by the classifiers when dealing with multiclassification problems. The value on the diagonal represents the number of correctly classified images of each class. The darker the diagonal color, the better performance of the classifiers. In this paper, the prediction results are normalized.

*4.3. Contrasting Experimental Results and Analysis.* To evaluate the effectiveness of the classification network proposed in this paper, we compare the proposed neural network model with VGG-16 [21], ResNet34, GoogleNet [22], DenseNet121, MobileNet, ShuffleNet, ShuffleNet\_SK, and ShuffleNet\_SE. To compare the results more confidently, all models use the dataset in this paper and are trained in the same training environment. As shown in Table 2, this study compares the accuracy, precision, recall,



TABLE 2: Network comparison experimental data.

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
VGG-16	50.72 ± 2.12	45.63 ± 3.25	45.67 ± 1.02	45.07 ± 1.59
ResNet34	83.95 ± 4.02	84.88 ± 3.18	81.28 ± 4.51	82.81 ± 3.44
GoogleNet	53.72 ± 5.42	47.43 ± 4.77	51.73 ± 4.82	45.09 ± 5.03
DenseNet121	86.39 ± 1.45	87.00 ± 1.91	83.95 ± 2.62	85.17 ± 1.98
MobileNet	54.30 ± 1.57	65.12 ± 2.18	44.60 ± 1.69	43.45 ± 2.03
ShuffleNet	80.37 ± 2.06	79.90 ± 1.89	79.42 ± 1.58	79.60 ± 1.95
ShuffleNet_SK	81.23 ± 2.03	81.65 ± 1.64	79.88 ± 2.25	80.67 ± 1.83
ShuffleNet_SE	81.38 ± 1.95	81.76 ± 2.32	80.74 ± 1.87	81.16 ± 2.26

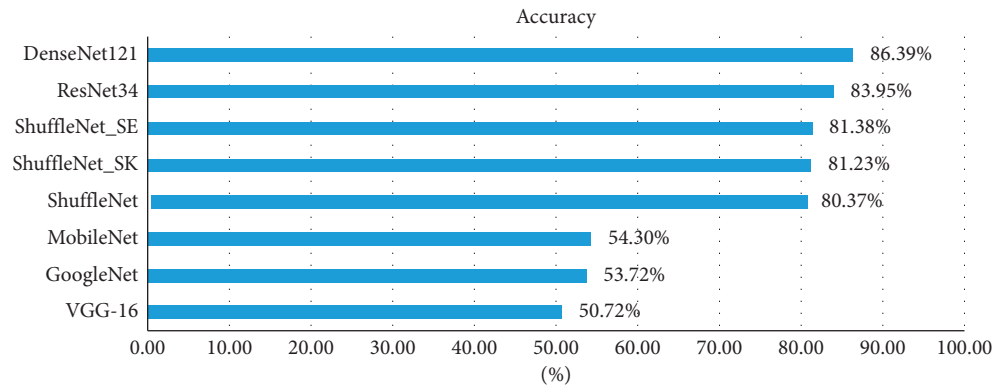


FIGURE 7: Accuracy comparison.

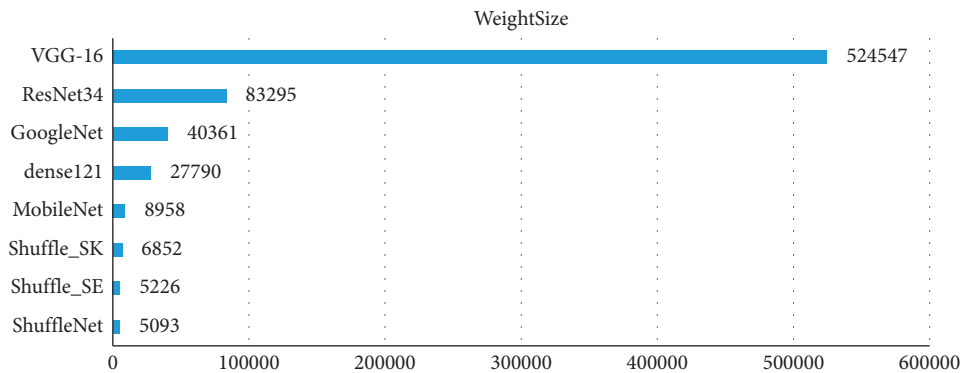


FIGURE 8: Model size comparison.

and F1-scores of the above networks. The mean and standard deviation were used to summarize the results. The results show that the classification ability of the improved network is significantly improved.

Figures 7 and 8 show that the model size of the improved network is greatly reduced compared with the traditional classification network and also greatly reduced compared with the lightweight network MobileNet. In terms of classification accuracy, the improved network maintains high recognition accuracy, and the classification performance is

improved compared with the unimproved ShuffleNet and significantly improved compared with MobileNet. Our model can improve computational efficiency significantly while achieving good performance in terms of classification accuracy, thus representing a reasonable balance between model size and performance.

As shown in Figure 9, the improved network may not be as effective as the network before improvement in one index, the prediction accuracy of the network model with SENet added is better.

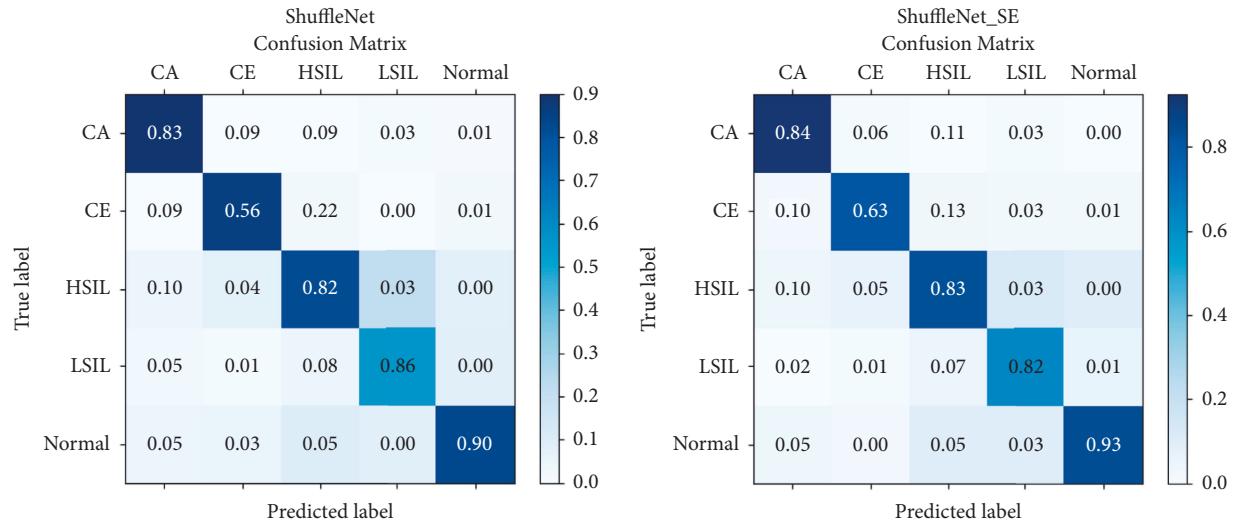


FIGURE 9: Confusion matrix.

## 5. Conclusions

This paper has proposed a dataset of colposcopic images using colposcopic images of cervical precancerous lesions and cervical cancer patients of different grades. We have also used six neural network models for comparative experiments and proposed two new deep learning-based lightweight network models ShuffleNet\_SK and ShuffleNet\_SE for multi-classification of cervical diseases. The classification performance is improved by adding attention on the inverted residual network. As a result, ShuffleNet\_SK and ShuffleNet\_SE achieved classification accuracy of 81.23% and 81.38%, respectively. The proposed networks are suitable for the mobile terminal with limited computing resources, which can classify cervical diseases more accurately and faster, so as to meet the demand of real-time, and has more practical clinical application value. Additionally, they can also be applied to prescreen for other types of cancer, reducing missed detection by physicians.

## Data Availability

The data are publicly available at [https://github.com/AluminiumOxide/ShuffleNet\\_Attention\\_Extend](https://github.com/AluminiumOxide/ShuffleNet_Attention_Extend).

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This study was supported by Baoding Science and Technology Planning Project (2141ZF306, 2141ZF135), and Supported by Foundation of President of Hebei University (XZJJ201918).

## References

[1] H. Sung, J. Ferlay, R. L. Siegel et al., "Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality

worldwide for 36 cancers in 185 countries," *CA: A Cancer Journal for Clinicians*, vol. 71, no. 3, pp. 209–249, 2021.

- [2] W. Hua, T. Xiao, X. Jiang et al., "Lymph-vascular space invasion prediction in cervical cancer: exploring radiomics and deep learning multilevel features of tumor and peritumor tissue on multiparametric MRI," *Biomedical Signal Processing and Control*, vol. 58, Article ID 101869, 2020.
- [3] T. I. Yusufaly, K. Kallis, A. Simon et al., "A knowledge-based organ dose prediction tool for brachytherapy treatment planning of patients with cervical cancer," *Brachytherapy*, vol. 19, no. 5, pp. 624–634, 2020.
- [4] V. Chandran, M. G. Sumithra, A. Karthick et al., "Diagnosis of cervical cancer based on ensemble deep learning network using colposcopy images," *BioMed Research International*, vol. 202115 pages, Article ID 5584004, 2021.
- [5] Q. Ji, J. Engel, and E. Craine, "Texture analysis for classification of cervix lesions," *IEEE Transactions on Medical Imaging*, vol. 19, no. 11, pp. 1144–1149, 2000.
- [6] F. Stübs, G. Mehlhorn, P. Gass et al., "Accuracy of colposcopy-directed biopsy in detecting early cervical neoplasia," *Geburtshilfe und Frauenheilkunde*, vol. 78, no. 10, pp. 193–194, 2018.
- [7] L. Denny, L. Kuhn, M. De Souza, A. E. Pollack, W. Dupree, and T. C. Wright, "Screen-and-Treat Approaches for Cervical Cancer Prevention in Low-Resource Settings: a randomized controlled trial," *JAMA*, vol. 294, no. 17, pp. 2173–2181, 2005.
- [8] R. Sankaranarayanan, B. M. Nene, S. S. Shastri et al., "HPV screening for cervical cancer in rural India," *New England Journal of Medicine*, vol. 360, no. 14, pp. 1385–1394, 2009.
- [9] R. Catarino, S. Schäfer, P. Vassilakos, P. Petignat, and M. Arbyn, "Accuracy of combinations of visual inspection using acetic acid or lugol iodine to detect cervical precancer: a meta-analysis," *BJOG: An International Journal of Obstetrics & Gynaecology*, vol. 125, no. 5, pp. 545–553, 2018.
- [10] N. Wentzensen, J. L. Walker, M. A. Gold et al., "Multiple biopsies and detection of cervical cancer precursors at colposcopy," *Journal of Clinical Oncology*, vol. 33, no. 1, pp. 83–89, 2015.
- [11] M. Suriya, V. Chandran, and M. G. Sumithra, "Enhanced deep convolutional neural network for malarial parasite classification," *International Journal of Computers and Applications*, pp. 1–10, 2019.

- [12] B. Sridhar, "Investigations of medical image segmentation methods with inclusion mathematical morphological operations," *Traitement du Signal*, vol. 38, no. 5, pp. 1531–1540, 2021.
- [13] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [14] A. Çınar, M. Yıldırım, and Y. Eroğlu, "Classification of pneumonia cell images using improved ResNet50 model," *Traitement du Signal*, vol. 38, no. 1, pp. 165–173, 2021.
- [15] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4700–4708, Honolulu, HI, USA, July 2017.
- [16] A. G. Howard, M. Zhu, B. Chen et al., "Mobilenets: efficient convolutional neural networks for mobile vision applications," 2017, <https://arxiv.org/abs/1704.04861>.
- [17] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: an extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6848–6856, Salt Lake City, UT, USA, June 2018.
- [18] F. Chollet, "Xception: deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1251–1258, Honolulu, HI, USA, July 2017.
- [19] J. Hu, L. Shen, S. Albanie, G. Sun, and E. H. Wu, "Squeeze-and-Excitation networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 8, pp. 2011–2023, 2020.
- [20] X. Li, W. Wang, X. Hu, and J. Yang, "Selective kernel networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 510–519, Manhattan, New York, US, June 2019.
- [21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, <https://arxiv.org/abs/1409.1556>.
- [22] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9, Boston, MA, USA, June 2015.

## Research Article

# Arrhythmia Classification Algorithm Based on a Two-Dimensional Image and Modified EfficientNet

Cui-fang Zhao , Wan-yun Yao , Mei-juan Yi , Chao Wan , and Yong-le Tian 

*College of Physics and Electronic Information Engineering, Zhejiang Normal University, Jinhua 321004, China*

Correspondence should be addressed to Cui-fang Zhao; [xx98@zjnu.cn](mailto:xx98@zjnu.cn)

Received 1 June 2022; Revised 9 July 2022; Accepted 22 July 2022; Published 27 August 2022

Academic Editor: Mohammed A. A. Al qaness

Copyright © 2022 Cui-fang Zhao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The classification and identification of arrhythmias using electrocardiogram (ECG) signals are of great practical significance in the early prevention and diagnosis of cardiovascular diseases. In this study, we propose an arrhythmia classification algorithm based on two-dimensional (2D) images and modified EfficientNet. First, we developed a method for converting original one-dimensional (1D) ECG signals into 2D image signals. In contrast with the existing classification method that uses only the time-domain features of a 1D ECG signal, the classification of 2D images can consider the spatiotemporal characteristics of the signal. Then, to better assign feature weights, we introduced an attention feature fusion module (AFF) into the EfficientNet network to replace the addition operation in the mobile inverted bottleneck convolution (MBConv) structure of the network. We selected EfficientNet for modification because, compared with most convolutional neural networks (CNNs), EfficientNet does not require manual adjustment of parameters, which improves the accuracy and speed of the network. Finally, we combined the 2D images and the improved EfficientNet network and tested its performance as an arrhythmia classification method. Our experimental results show that the network training of the proposed method requires less equipment and training time, and this method can effectively distinguish eight types of heartbeats in the MIT-BIH arrhythmia database, with a classification accuracy of 99.54%. Thus, the model has a good classification effect.

## 1. Introduction

The 2020 report on cardiovascular health and disease in China shows that the incidence and mortality of cardiovascular disease have been increasing, while the age of onset has been decreasing [1]. Arrhythmia is an important group of cardiovascular diseases, and its early detection plays a crucial role in the treatment of cardiovascular diseases. The diagnosis of arrhythmia mostly depends on the electrocardiogram (ECG), and the classification of arrhythmia by analyzing the ECG has become a hot research topic [2].

The traditional classification and identification of arrhythmia rely mainly on extracting features [3] such as timing features, statistical features, and morphological features [4–7]. The QRS complex, the most widely utilized feature in the field, is generally processed by employing Hermite polynomials, wavelet transforms, high-order statistics, and other techniques before extracting morphological

characteristics [5–13]. Because of the emergence of deep learning, researchers often use neural network feature selection instead of manual feature selection to achieve automatic feature extraction [14–25]. Hannun et al. [19] directly input the one-dimensional (1D) ECG signal into the improved ResNet-34 deep learning network for the first time, realizing end-to-end arrhythmia classification. Lu et al. [24] used the convolution method to convert the 1D ECG signal into a two-dimensional (2D) image for the first time, and they fused temporal features for the classification of five types of arrhythmia, with an accuracy rate of 99%. Huang et al. [20] converted ECG signals into time-spectrograms through a short-time Fourier transform, and used 2D convolutional neural networks (CNNs) to classify five types of arrhythmias, achieving an accuracy of 99%. Compared with the 1D training accuracy of 90.93%, the 2D image training effect was better. Naz et al. [23] converted ECG signals into  $32 \times 32$  binary images and used several deep

CNNs for ventricular tachyarrhythmia recognition, achieving an accuracy of 97.6%. Maskeliunas et al. [25] obtained the statistical features extracted from 2D images based on Gramian angular field (GAF), achieving 86% accuracy in the classification of premature ventricular contraction (PVC) beats versus normal (NOR) beats. Akbar et al. [17] extracted ECG signal features based on time-spectral entropy and input them into the CNN, realizing the classification of five types of arrhythmias, and the classification accuracy reached 98.33%. In the same year, Min et al. [21] used the GAF transformation to convert 1D signals into 2D signals and then utilized the transfer CNN to achieve the classification of five types of arrhythmias. The above methods convert 1D ECG signals into 2D signals and use neural networks to classify them with good results, but manual parameter tuning is required in most CNN structures, the number of iterations is high, the process is complex and time-consuming, equipment requirements are high, there are few types of arrhythmias involved in arrhythmia classification, and there is still room to improve the classification accuracy.

To improve classification accuracy, the width, depth, or resolution of the network is generally increased. Although two or three dimensions can be arbitrarily scaled, arbitrary scaling requires tedious manual tuning, and usually still produces sub-optimal accuracy and efficiency. Tan et al. [26] proposed EfficientNet, whose main idea is to search for an efficient baseline and then use hybrid scaling, which combines depth, width, and resolution scaling according to certain rules. It has few network parameters and a much higher speed while providing good accuracy, which improves the practicality of the network as well as the industrial landing possibility. Through transfer learning, the EfficientNet network has achieved a good level of performance on several well-known data sets and good results in medicine [27–30]. Feature fusion is usually implemented with simple linear operations. Attention feature fusion (AFF) [31] can better fuse semantically inconsistent and scale-inconsistent features and is suitable for short connections.

In this study, we developed a preprocessing method to convert 1D ECG signals into 2D images and modified the EfficientNet network to achieve arrhythmia classification. We selected EfficientNet for modification due to its transfer learning capabilities, fast training speed, and high efficiency and because it does not require manual adjustment of network parameters. The aim of the modification was to improve EfficientNet and apply it to arrhythmia classification. The novelty and contribution of this study are as follows:

- (1) A preprocessing method is proposed to convert the original 1D ECG signal into a 2D image, which reflects the spatiotemporal features of the signal.
- (2) AFF is introduced to replace the addition operation in the MBConv structure of the EfficientNet network.

- (3) The proposed method effectively distinguishes eight types of heartbeats in the MIT-BIH arrhythmia database, with a classification accuracy of 99.54%.

## 2. Materials and Methods

In this section, we will briefly introduce the database that we used for ECG classification and describe our data preprocessing and network. The flow diagram of the proposed method is shown in Figure 1.

*2.1. Database.* We obtained the experimental data in this study from the MIT-BIH arrhythmia database, which has approximately 110,000 ECG beats, including 16 different types of arrhythmias [32, 33]. Considering that unclassifiable beats such as paced heartbeats in the database have usually been ignored in prior ECG arrhythmia classification studies, we selected eight common arrhythmias for classification in this study: NOR, PVC, paced beat (PAB), right bundle branch block beat (RBBB), left bundle branch block beat (LBBB), atrial premature contraction (APC), ventricular flutter wave (VFW), and ventricular escape beat (VEB). The selected data codes and sample numbers are shown in Table 1.

*2.2. Preprocessing.* ECG signal preprocessing mainly refers to beat segmentation and signal filtering. Generally, a relatively complete ECG signal includes at least the  $P$  wave, QRS complex, and  $T$  wave, and the time intervals of each waveband are shown in Table 2 [34]. It can be seen from Table 2 that the minimum duration of the complete  $P$  wave, QRS complex, and  $T$  wave is 0.44 s (including the  $P$ - $R$  interval and  $Q$ - $T$  interval). Combined with the sampling frequency of 360 Hz, a relatively complete heartbeat sequence length of at least 158 sample points can be obtained. Figure 2(a) shows the original waveform of an ECG signal, whose horizontal coordinate is the number of sampling sequence points, and the vertical coordinate is the amplitude of the ECG signal. We took the  $R$ -peak marked by the expert as the dividing point and extended point  $m1$  to the left (including the  $R$ -peak point), extended point  $m2$  to the right (excluding the  $R$ -peak point), and performed dynamic segmentation to form a cardiac slice, which contained a relatively complete  $P$  wave, QRS complex, and  $T$  wave. Then, we made a single heartbeat ECG signal waveform diagram, setting  $A = \{A_1, A_2, \dots, A_t\}$ , as shown in Figure 2(b), where the displayed abscissa range is  $[n_{\min}, n_{\max}]$ , and the ordinate range is  $[A_{\min}, A_{\max}]$ . To unify the distribution of beats, we normalized the beats and converted them into 2D images. We set the amplitude of the ECG signal as  $\{Af_n, n \in [1, m1 + m2]\}$  and the pixel value of the 2D image as  $\{I_{a,b}, a \in [1, 224], b \in [1, 224]\}$ . The corresponding point relationship of the conversion from the 1D ECG signal to the 2D image signal is shown in Figure 2(b), and the conversion formulas are as follows:

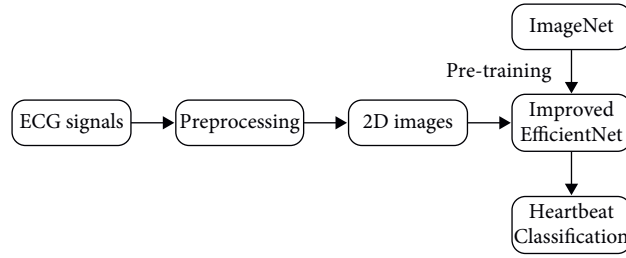


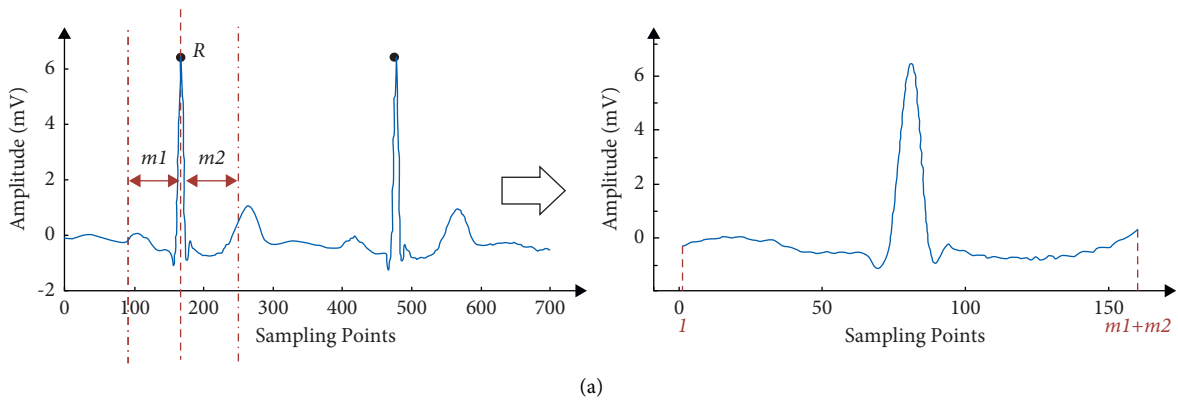
FIGURE 1: Flow diagram of the proposed method.

TABLE 1: MIT-BIH arrhythmia database.

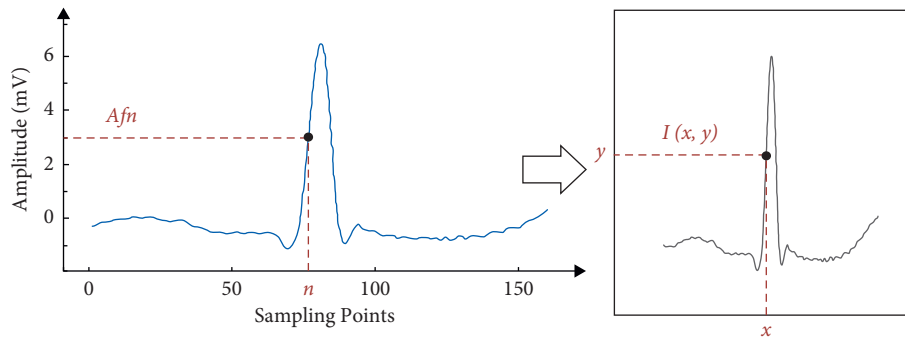
Type	Records	Number of beats
NOR	100, 101, 103, 105, 108, 112, 113, 114, 115, 117, 121, 122, 123, 202, 205, 219, 230, 234	75,016
PVC	106, 116, 119, 200, 201, 203, 208, 210, 213, 215, 221, 228, 233	7,130
PAB	102, 104, 107, 217	7,024
RBBB	118, 124, 212, 231	7,256
LBBB	109, 111, 207, 213	8,072
APC	209, 220, 222, 223, 232	2,544
VFW	207	472
VEB	207	106
<b>Total</b>		<b>107,620</b>

TABLE 2: Time interval table of each waveband of the ECG signal.

Wave	<i>P</i>	<i>P-R</i>	QRS	<i>Q-T</i>	<i>T</i>
Time interval (s)	0.12	0.12–0.20	0.06–0.10	0.32–0.44	0.05–0.25



(a)



(b)

FIGURE 2: Data preprocessing: (a) heartbeat interception: extract a heartbeat from the original signal and (b) 1D to 2D: transform a 1D signal into a 2D image.

```

(i) Algorithm
(ii) Input:  $A = \{A_1, A_2, \dots, A_t\}$ ,  $m1$ ,  $m2$ 
(iii) Output:  $I = \{I_1(a, b), \dots, I_N(a, b)\}$ 
(iv) Begin
(v) Find  $R$ -peak points:  $R_{\text{index}}[N]$ 
(vi) for  $j = 1$  to  $N$  do
(vii) Heartbeat interception:  $A' \leftarrow A[R_{\text{index}}[j] - m1 : R_{\text{index}}[j] + m2]$ 
(viii) The 1D signal to the 2D image: according to equations (1)–(3), get  $I_j(a, b)$ 
(ix) Return I

```

ALGORITHM 1: Pseudocode of proposed preprocessing method.

$$I(a, b) = \begin{cases} 255 * \frac{Af_n - \min(Af_n)}{\max(Af_n) - \min(Af_n)} & a = x \& b = y \\ 255 & \text{else} \end{cases}, \quad (1)$$

$$x = 224 * \frac{n - n_{\min}}{n_{\max} - n_{\min}}, \quad (2)$$

$$y = 224 * \frac{Af_n - A_{\min}}{A_{\max} - A_{\min}}. \quad (3)$$

Among these formulas, (1) returns the pixel expressions for 2D images, (2) is the formula for transforming the horizontal coordinates of 2D images, and (3) is the formula for transforming the vertical coordinates of 2D images. The pseudocode of the proposed preprocessing method is shown in Algorithm 1.

There is a lot of noise in the process of ECG signal acquisition owing to the influence of the in vivo and in vitro environment; thus, we used the morphological corrosion function to denoise the 2D images to create a heartbeat sample.

### 3. Improved EfficientNet

EfficientNet features several distinct network models ranging from B0 to B7, with B1 to B7 continuously increasing the number of layers, parameters, and sub-blocks based on B0. Simultaneously, the resolution of the input image is growing, which means that equipment needs are continually increasing. Here, we chose the EfficientNet-B0 network as the classification model based on the features included in the 2D images of the cardiac slices and the hardware resources of the available equipment. The network input image resolution requirement for EfficientNet-B0 is  $224 \times 224$ . As shown in the analysis in Section 2.2, transforming the waveform picture into an image with a resolution of  $224 \times 224$  meets the requirements.

The MBCConv module is the core structure of the EfficientNet-B0 network. Its structural diagram is shown in Figure 3(a). A simple addition operation is used to realize the feature fusion of different branches. In this study, we introduced the AFF [31] to replace the addition operation in the MBCConv structure so as to better allocate the weight of

the features. The improved MBCConv structure is shown in Figure 3(b), and the mapping relationship of AFF is

$$Z = M(X \cup Y) \otimes X + (1 - M(X \cup Y)) \otimes Y, \quad (4)$$

where  $\cup$  is the initial feature fusion of the two input features ( $X$  and  $Y$ ),  $M$  is the multi-scale attention module function, and  $\otimes$  represents feature multiplication.

## 4. Results and Analysis

To validate the effectiveness of the algorithm, we performed model training on Intel CPU, NVIDIA GTX1650 GPU, using the Python-based PyTorch framework with PyTorch 1.9 and Python 3.8. We randomly divided the ECG signal image data set into two parts: the training set of 96,858 images, 20% of which we designated as the validation set (19,371 images), and the test set of 10,762 images.

*4.1. Heartbeat Sequence Length Comparison Experiment.* From Section 2.2, we know that a more complete heartbeat sequence containing a complete  $P$  wave, QRS complex, and  $T$  wave is at least 158 sample points in length. When the intercepted heartbeat sequence is too short, the sample will contain insufficient information, and the classification accuracy will be low; when the sequence is too long, the sample will contain a large amount of information, but some information may be redundant, so there is little room for improving the classification accuracy, and the time it takes to initialize data increases.

In this study, we used varied sequence lengths ( $L$ ) for related experiments to verify the influence of heartbeat sequence length on classification performance. We defined  $L$  as 130, 160, 170, 180, 190, 200, and 250 sample points, with corresponding left and right extension points ( $m1$  and  $m2$ ) satisfying  $m1 = m2 = 65$ ,  $m1 = m2 = 80$ ,  $m1 = m2 = 85$ ,  $m1 = m2 = 90$ ,  $m1 = m2 = 95$ ,  $m1 = m2 = 100$ , and  $m1 = 100$  and  $m2 = 150$ , respectively. Figure 4(a) depicts the accuracy of the training set versus the number of epochs of model training for the different sequence lengths. The accuracy of the training set under different sequence lengths is largely stable after 300 epochs. We chose the best model for testing, and Figure 4(b) shows the accuracy comparison graph of the test set over various lengths. As  $L$  increases, the accuracy of

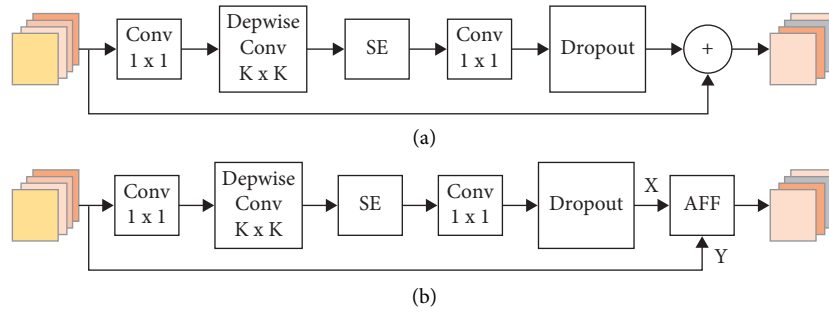


FIGURE 3: Comparison before and after the improvement of the MBConv structure: (a) the structure of MBConv and (b) the improved structure of MBConv, wherein AFF replaces the addition operation in the MBConv structure.

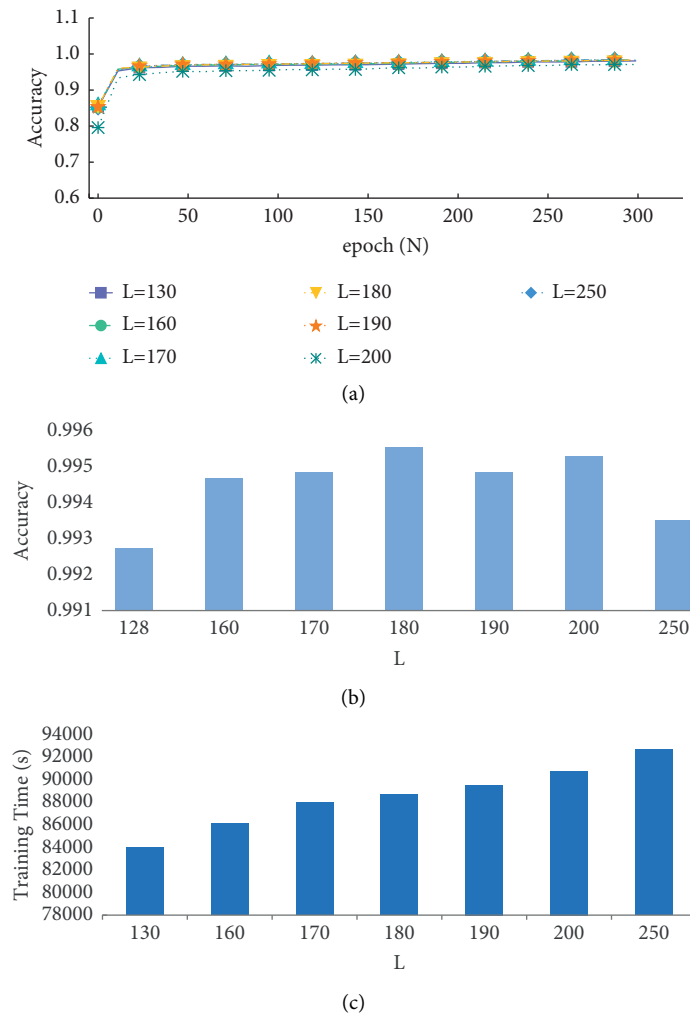


FIGURE 4: Comparison of accuracy for different heartbeat sequence lengths ( $L$ ): (a) comparison of training set accuracy corresponding to different heartbeat sequence lengths, (b) comparison of test set accuracy corresponding to different heartbeat sequence lengths, and (c) comparison of training time corresponding to different heartbeat sequence lengths.

the test set increases from 130, peaks at 180, and then declines. This suggests that the duration of the heartbeat sequence influences accuracy and different sequence lengths contain varied ECG signal properties. Although the physical properties of the QRS complex are the most important for

defining the heart rate type, the  $P$  wave and  $T$  wave also have an impact on the classification outcomes.  $L = 130$  contains mostly the QRS complex and has an incomplete  $P$  wave and  $T$  wave;  $L = 160-200$  includes a more complete  $P$  wave, QRS complex, and  $T$  wave with more features; and  $L = 250$



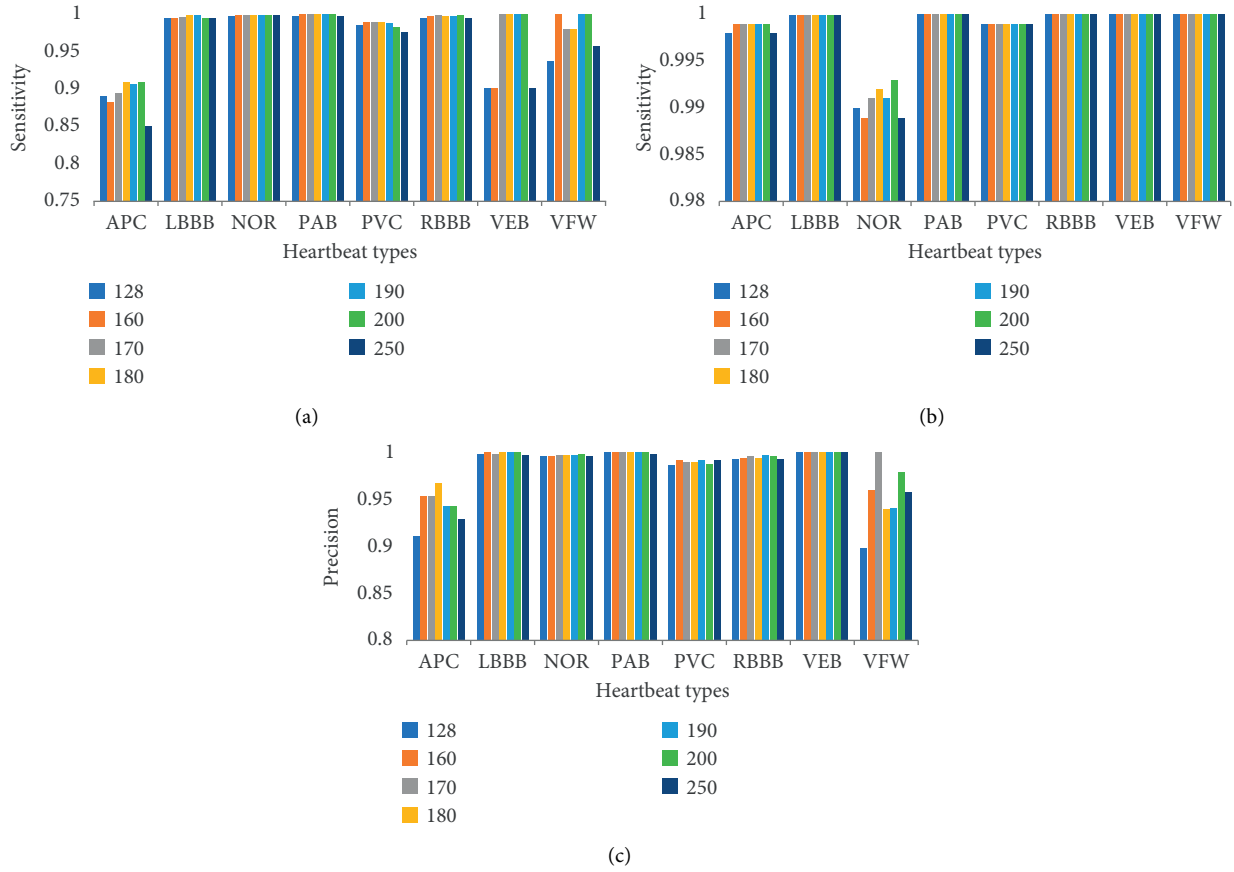


FIGURE 5: Comparison of test results for different heartbeat sequence lengths ( $L$ ): (a) comparison of test sensitivity corresponding to different heartbeat sequence lengths, (b) comparison of test specificity corresponding to different heartbeat sequence lengths, and (c) comparison of test precision corresponding to different heartbeat sequence lengths.

includes not only a complete  $P$  wave, QRS complex, and  $T$  wave but also a gentle wave before and after the  $P$  wave and after the  $T$  wave. Too many sample points make features less distinct while also causing accuracy swings. Additionally, as seen in Figure 4(c), more sample points equate to a longer training time.

Considering the inevitable data imbalance in the medical data set sample, we employed evaluation metrics such as sensitivity, specificity, and precision [20] for further comparison, with the results displayed in Figure 5. When comparing the evaluation indexes in Figure 5, it can be seen that the total index of  $L=180$  is better than that of other examples, particularly in APC, where the precision rate and sensitivity are greatly increased. The type of arrhythmia is more accurate in general.

**4.2. Experiment on Classification of EfficientNet Network before and after Improvement.** To verify the performance of the classification method based on the improved EfficientNet network, we conducted experiments on a sample library of heartbeats with a sequence length  $L=180$ . To ameliorate the data imbalance problem in the medical data set samples, we performed data augmentation for two types of samples: VEB and VFW; we added four data sets with the

heartbeat sequence lengths  $L=160$ ,  $L=170$ ,  $L=190$ , and  $L=200$  to the data set of  $L=180$ .

Table 3 shows the outcomes of EfficientNet before and after improvement, as well as the classification evaluation indexes before and after data augmentation. Compared with other methods, AFF-EfficientNet-B0 + data augmentation has significantly higher accuracy for APC and VFW compared to the other three methods, with an overall accuracy of 99.54. APC is a common clinical arrhythmia with symptoms such as palpitations, and some people may be asymptomatic; VFW is more commonly observed in those with serious heart problems and is diagnosed mostly through an ECG examination. Thus, the results indicate that the revised model enhances APC and VFW recognition accuracy and makes follow-up treatment easier.

The classification confusion matrix of AFF-EfficientNet-B0 + data augmentation is given in Figure 6. Figure 6 shows that the categorization accuracy of all eight ECG signal types is relatively high, and there is relatively more confusion between the three categories of APC, PVC, and NOR.

Figure 7 shows the typical correct sample heartbeat maps, and Figure 8 shows some of the misidentified sample heartbeat maps. The main reason for the misclassification of APC as NOR in Figure 8(a) is that this APC heartbeat picture

TABLE 3: Comparison of sensitivity, specificity, and precision at  $L = 180$ .

Evaluation indicator	Method	Type							
		APC	LBBB	NOR	PAB	PVC	RBBB	VEB	VFW
Sensitivity	Method 1	0.89	0.995	0.998	1	0.989	0.997	1	0.979
	Method 2	0.906	0.996	0.998	1	0.99	0.999	1	0.987
	Method 3	0.909	0.998	0.998	1	0.989	0.997	1	0.979
	Method 4	0.902	0.998	0.999	1	0.987	0.997	1	0.991
Specificity	Method 1	0.89	0.995	0.998	1	0.989	0.997	1	0.979
	Method 2	0.906	0.996	0.998	1	0.99	0.999	1	0.987
	Method 3	0.909	0.998	0.998	1	0.989	0.997	1	0.979
	Method 4	0.902	0.998	0.999	1	0.987	0.997	1	0.991
Precision	Method 1	0.954	1	0.996	1	0.99	0.994	1	0.958
	Method 2	0.962	1	0.997	1	0.986	0.993	1	0.979
	Method 3	0.967	1	0.996	1	0.989	0.994	1	0.939
	Method 4	0.974	1	0.996	1	0.989	0.993	1	0.987

Notes: Method 1, Method 2, Method 3, and Method 4 represent EfficientNet-B0, EfficientNet-B0 + data augmentation, AFF-EfficientNet-B0, and AFF-EfficientNet-B0 + data augmentation, respectively.

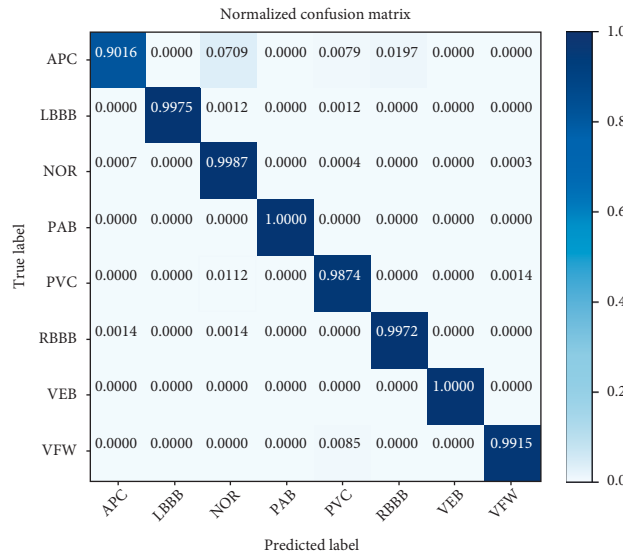


FIGURE 6: Confusion matrix representation of the test classification results for  $L = 180$ .

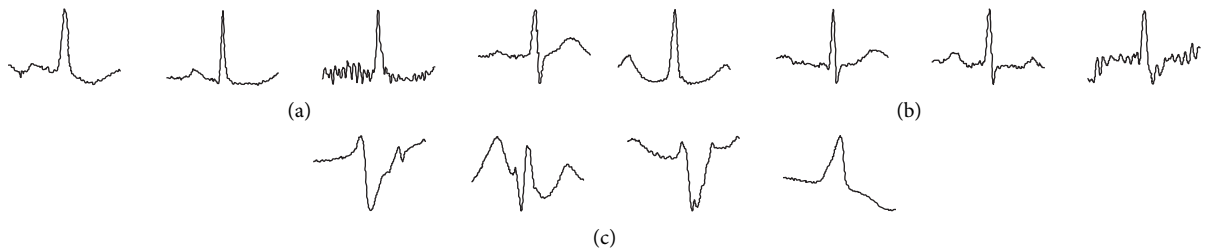


FIGURE 7: Typical correct sample heartbeat maps. (a) NOR. (b) APC. (c) PVC.

has a large difference at the  $P$  wave compared to most of the samples in the APC library, and the reason for the misclassification of NOR as APC in Figure 8(b) is that in addition to the  $P$  wave, there is also a large undulation at its  $T$  wave, which is different from the normal NOR sample in Figure 7.

In addition, the heartbeat of the same patient generally has the greatest similarity in terms of APC and NOR; for example, patient number 100 was recorded to have both APC and NOR, so this also contributed to some extent to the result that the model sometimes confused APC and NOR. In Figure 8(c), the NOR sample mistakenly detected as a PVC

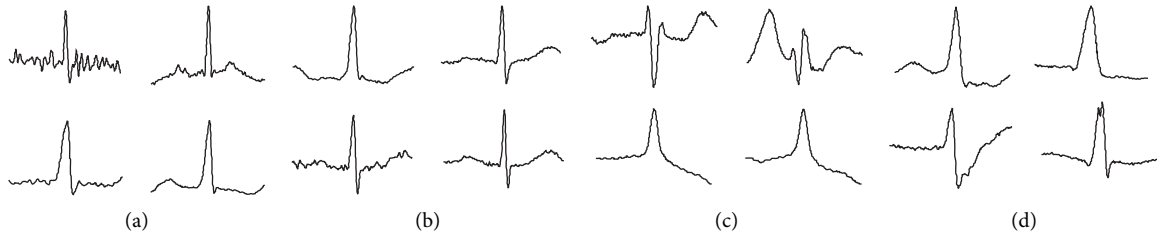


FIGURE 8: Some misclassified sample heartbeat maps: (a) heartbeat images misclassified as NOR instead of APC, (b) heartbeat images misclassified as APC instead of NOR, (c) heartbeat images misclassified as PVC instead of NOR, and (d) heartbeat images misclassified as NOR instead of PVC.

TABLE 4: Comparison of different methods.

Author	Method	Type	Accuracy (%)	Number of samples
Plawiak and Acharya [35]	Welch method and discrete Fourier transform	17	95.00	744
Liu et al. [35]	Wavelet scattering transform	4	99.30	100,507
Yang et al. [15]	ECG morphological parameters and visual pattern characteristics	15	97.70	104,986
Romdhane et al. [36]	Building a deep CNN model	5	98.41	109,446
This paper	1D to 2D + AFF-EfficientNet	8	99.54	107,620

sample was similar to the normal PVC sample in Figure 7, which was an atypical case in the NOR sample pool. The same was true for the PVC sample misclassified as a NOR sample in Figure 8(d).

The analysis of the classification accuracy by sequence length and the analysis of the misclassified samples indicate there are obvious shortcomings in existing methods that use the QRS complex as the main feature for ECG signal classification, and the small undulations on both sides of the QRS complex have some influence on the classification accuracy. Thus, the classification based solely on the QRS complex is not ideal.

**4.3. Comparison Experiments with Other Classification Algorithms.** Table 4 shows the results of comparing the algorithm in this paper with other algorithms, all using the MIT-BIH arrhythmia database. Plawiak and Acharya [35] used 10 s as the time base to intercept ECG signal samples to build a database, employed the discrete Fourier transform to extract features, and combined the features with the CNN for classification and identification. With a small number of classification samples, the accuracy reached 95%. Liu et al. [35] used heartbeats from a database established by baseline intercepted samples and extracted the signal features under eight different time windows using a wavelet scattering transform. After using principal component analysis (PCA) for dimensionality reduction and a  $k$ -nearest neighbor classifier for four classifications, they obtained an accuracy of 99.3%. Yang et al. [15] extracted ECG morphological parameters, such as amplitude, time interval, and QRS complex morphological features and combined them with the  $k$ -nearest neighbor classifier for 15 classifications, achieving an accuracy of 97.7%. Romdhane et al. [36] used the focal loss to construct a novel loss function and optimized the CNN

model to achieve five end-to-end classifications with an accuracy of 98.41%. Several recent papers have the problem of data imbalance, which is the norm for medical data sets.

In this study, eight categories of classification were achieved, with an accuracy of 99.54%, which was higher than the accuracy of existing arrhythmia classification methods, but the same problem of sample imbalance existed; increasing samples and evaluation metrics such as specificity and sensitivity were used to mitigate this problem. In comparison to existing methods, the accuracy of the method proposed in this study was found to be higher, and it achieved the classification of more categories. In addition, the proposed method had low equipment requirements and fast training and testing time. The average training time for each epoch was 4.5 min; the single test time was 0.0027 s; and the model size was 16,713 KB.

## 5. Discussion and Conclusion

In this study, we developed a method for converting original 1D ECG signals into 2D image signals. To better assign feature weights, we introduced AFF to replace the addition operation in the MBConv structure of the EfficientNet network.

The main limitation of the proposed arrhythmia classification algorithm is the low positive prediction accuracy for identifying APC beats. This is caused by data imbalance: specifically, there are many more NOR beats than other beats. The ratio of APC beats is only 2.3% in the data set. Moreover, multiple ECG samples from the same patient will generally exhibit the greatest similarity in heartbeats. The study results of the data augmentation show that the positive prediction accuracy for identifying VEB is substantially increased and ranges from 97.9% to 99.1%.

Given the influence of available laboratory equipment, we converted 1D ECG signals into 2D image signals and used spatiotemporal characteristics to perform classification experiments on eight ECG signal types in the MIT-BIH arrhythmia database, achieving relatively high accuracy of 99.54% based on the improved EfficientNet-B0 network. Most medical data sets have sample imbalance problems, which are generally mitigated by increasing a few types of samples or decreasing most types of samples. In this study, we applied the preprocessing method of 1D to 2D ECG signal conversion, which increased the amount of data, and selected the best length. Additionally, we performed data augmentation for two categories, VEB and VFW, and we added four similar groups of different-length images to this data set, which alleviated the data imbalance problem to some extent. Finally, we employed three evaluation indices, namely, sensitivity, specificity, and precision rate ground, to evaluate the model's effect, all of which were found to be high, indicating that the model has a good classification effect.

To extend the sample, the next step will be to identify relevant volunteers for sample collection. Validation of more ECG signal databases will be considered in the future to improve the practicality and robustness of the classification method for eventual application on medical robots or ECG signal monitoring devices. This approach can help doctors more accurately and quickly diagnose cardiovascular diseases from ECG signals [37].

## Data Availability

Publicly available data sets were analyzed in this study. These data sets can be found at <https://archive.physionet.org/physiobank/database/html/mitdbdir/mitdbdir.htm>.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This research was partially funded by the National Natural Science Foundation of China (no. 42075140) and the Public Welfare Fund of Zhejiang Province (no. LGF20D050004).

## References

- [1] Li. Xi, Wu. Chaoqun, Lu. Jiapeng, C. Bowang, Li. Yichong, and Y. Yang, "China cardiovascular health and disease report 2020 compilation group summary of China cardiovascular health and disease report 2020," *China Cardiovascular Research*, vol. 19, no. 07, pp. 582–590, 2021.
- [2] P. De Chazal and R. B. Reilly, "A patient-adapting heartbeat classifier using ECG morphology and heartbeat interval features," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 12, pp. 2535–2543, 2006.
- [3] M. Wasimuddin, K. Elleithy, A. S. Abuzneid, M. Faezipour, and O. Abuzagheh, "Stages-based ECG signal analysis from traditional signal processing to machine learning approaches: a survey," *IEEE Access*, vol. 8, Article ID 177782, 2020.
- [4] L. Chun-Cheng, Chun-Min, and Yang, "Heartbeat Classification Using Normalized RR Intervals and Morphological Features," *Mathematical Problems in Engineering Theory Methods & Applications*, Article ID 712474, 2014.
- [5] K. S. Park, B. H. Cho, and D. H. Lee, "Hierarchical support vector machine based heartbeat classification using higher order statistics and hermite basis function," *Computers in Cardiology IEEE*, 2008.
- [6] S. M. Anwar, M. Gul, M. Majid, and M. Alnowami, "Arrhythmia classification of ECG signals using hybrid features," *Computational and Mathematical Methods in Medicine*, pp. 1–8, Article ID 1380348, 2018.
- [7] P. deChazal, M. O'Dwyer, and R. B. Reilly, "Automatic classification of heartbeats using ECG morphology and heartbeat interval features," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 7, pp. 1196–1206, 2004.
- [8] C. Ye, B. V. K. V. Kumar, and M. T. Coimbra, "Combining General Multi-Class and Specific Two-Class Classifiers for Improved Customized ECG Heartbeat Classification," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, IEEE, Tsukuba, Japan, November 2012.
- [9] Z. Zhang, J. Dong, X. Luo, K. S. Choi, and X. Wu, "Heartbeat classification using disease-specific feature selection," *Computers in Biology and Medicine*, vol. 46, no. 1, pp. 79–89, 2014.
- [10] T. Ince, S. Kiranyaz, and M. Gabbouj, "A generic and robust system for automated patient-specific classification of ECG signals," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 5, pp. 1415–1426, 2009.
- [11] Y. Kutlu and D. Kuntalp, "Feature extraction for ECG heartbeats using higher order statistics of WPD coefficients," *Computer Methods and Programs in Biomedicine*, vol. 105, no. 3, pp. 257–267, 2012.
- [12] Y. Xiang, Z. Lin, and J. Meng, "Automatic QRS complex detection using two-level convolutional neural network," *BioMedical Engineering Online*, vol. 17, no. 1, p. 13, Jan 2018.
- [13] H. Yang and Z. Wei, "Arrhythmia recognition and classification using combined parametric and visual pattern features of ECG morphology," *IEEE Access*, vol. 8, Article ID 47103, 2020.
- [14] A. Majumdar and R. Ward, "Robust Greedy Deep Dictionary Learning for ECG Arrhythmia Classification," in *Proceeding of the International Joint Conference On Neural Networks*, pp. 4400–4407, Anchorage, AK, USA, 2017.
- [15] A. Sellami and H. Hwang, "A robust deep convolutional neural network with batch-weighted loss for heartbeat classification," *Expert Systems with Applications*, vol. 122, no. 3, pp. 75–84, 2019.
- [16] U. R. Acharya, S. L. Oh, Y. Hagiwara et al., "A deep convolutional neural network model to classify heartbeats," *Computers in Biology and Medicine*, vol. 89, pp. 389–396, 2017.
- [17] A. Asgharzadeh-Bonab, M. C. Amirani, and A. Mehri, "Spectral entropy and deep convolutional neural network for ECG beat classification," *Biocybernetics and Biomedical Engineering*, vol. 40, no. 2, pp. 691–700, Apr 2020.
- [18] S. M. Mathews, C. Kambhamettu, and K. E. Barner, "A novel application of deep learning for single-lead ECG classification," *Computers in Biology and Medicine*, vol. 99, pp. 53–62, Aug 2018.
- [19] A. Y. Hannun, P. Rajpurkar, M. Haghpanahi et al., "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network," *Nature Medicine*, vol. 25, no. 1, pp. 65–69, 2019.

- [20] J. Huang, B. Chen, B. Yao, and W. He, "ECG arrhythmia classification using STFT-based spectrogram and convolutional neural network," *IEEE Access*, vol. 7, Article ID 92871, 2019.
- [21] M. Chen and R. Wang, "Arrhythmia classification based on two-dimensional image and migration convolution neural network," *Computer Engineering*, vol. 46, no. 10, pp. 315–320, 2020.
- [22] V. J. Prakash and N. K. Karthikeyan, "Dual-layer deep ensemble techniques for classifying heart disease," *Information Technology and Control*, vol. 51, no. 1, pp. 158–179, 2022.
- [23] M. Naz, J. H. Shah, M. A. Khan, M. Sharif, M. Raza, and R. Damasevicius, "From ECG signals to images: a transformation based approach for deep learning," *PeerJ Computer Science*, vol. 7, no. 1, p. e386, 2021.
- [24] W. Lu, H. Hou, and J. Chu, "Feature fusion for imbalanced ECG data analysis," *Biomedical Signal Processing and Control*, vol. 41, pp. 152–160, 2018.
- [25] R. Damasevicius, R. Maskeliunas, M. Wozniak, and P. Dawid, "Visualization of physiologic signals based on hjorth parameters and gramian angular fields," in *Proceedings of the 2018 IEEE 16th World Symposium on Applied Machine Intelligence and Informatics (SAMI) IEEE, Kosice and Herlany, Slovakia, February 2018*.
- [26] M. Tan and Q. V. Le, "EfficientNet: rethinking model scaling for convolutional neural networks," *Long Beach*, vol. 2019, Article ID 10691, 2019.
- [27] N. Siddique, S. Paheding, Z. Alom, and V. K. Devabhaktuni, "Recurrent residual U-Net with EfficientNet encoder for medical image segmentation," *Pattern Recognition and Tracking*, 2021.
- [28] A. Abedalla, M. Abdullah, M. Al-Ayyoub, and E. Benkhelifa, "Chest X-ray pneumothorax segmentation using U-Net with EfficientNet and ResNet architectures," *PeerJ Computer Science*, vol. 7, pp. e607–65, 2021.
- [29] G. Marques, D. Agarwal, and I. de la Torre Diez, "Automated medical diagnosis of COVID-19 through EfficientNet convolutional neural network," *Applied Soft Computing*, vol. 96, Article ID 106691, 2020.
- [30] I. U. Khan, N. Aslam, T. Anwar et al., "Remote diagnosis and triaging model for skin cancer using EfficientNet and extreme gradient boosting," *Complexity*, vol. 22, pp. 1–13, 2021.
- [31] Y. Dai, F. Gieseke, S. Oehmcke, Y. Wu, and K. Barnard, "Attentional Feature Fusion," pp. 3560–3569, 2021.
- [32] A. L. Goldberger, L. A. N. Amaral, L. Glass et al., *Circulation*, vol. 101, no. 23, pp. E215–E220, 2000.
- [33] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH arrhythmia database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.
- [34] G. X. Yan, R. S. Lankipalli, J. F. Burke, S. Musco, and P. R. Kowey, "Ventricular repolarization components on the electrocardiogram: cellular basis and clinical significance," *Journal of the American College of Cardiology*, vol. 42, no. 3, pp. 401–409, 2003.
- [35] P. Plawiak and U. R. Acharya, "Novel deep genetic ensemble of classifiers for arrhythmia detection using ECG signals," *Neural Computing & Applications*, vol. 32, no. 15, Article ID 11137, Aug 2020.
- [36] Z. Liu, G. Yao, Q. Zhang, J. Zhang, and X. Zeng, "Wavelet scattering transform for ECG beat classification," *Computational and Mathematical Methods in Medicine*, Article ID 3215681, 2020.
- [37] T. F. Romdhane, H. Alhichri, R. Ouni, and M. Atri, "Electrocardiogram heartbeat classification based on a deep convolutional neural network and focal loss," *Computers in Biology and Medicine*, vol. 123, Article ID 103866, 2020.

## Research Article

# A Comparison on LSTM Deep Learning Method and Random Walk Model Used on Financial and Medical Applications: An Example in COVID-19 Development Prediction

Yifan Yao <sup>1</sup>, Xinxin Li <sup>1</sup> and Qing Li <sup>2</sup>

<sup>1</sup>School of Fintech, Hebei Finance University, Baoding, China

<sup>2</sup>Finance Department, Capital University of Economics and Business, Beijing, China

Correspondence should be addressed to Xinxin Li; 208088@cueb.edu.cn

Received 14 January 2022; Accepted 26 July 2022; Published 23 August 2022

Academic Editor: Ahmed A. Ewees

Copyright © 2022 Yifan Yao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This study aims to establish the model of the cryptocurrency price trend based on a financial theory using the Long Short-Term Memory (LSTM) networks model with multiple combinations between the window length and the predicting horizons. The Random Walk model is also applied with different parameter settings. The object of this study is the cryptocurrency and medical issues, primarily the Bitcoin and Ethereum and the COVID-19. Quantitative analysis is adopted as the method of this dissertation. The research tool is Python programming language, and the TensorFlow package is employed to model and analyze research topics. The results of this study show the limitations of the LSTM and Random Walk model for price prediction while demonstrating the different characteristics of both models with different parameter settings, providing a balance between the model's accuracy and the model's practicality.

## 1. Introduction

The subject of general dynamics for digital currencies is a popular one in the literature of modern cryptocurrency analysis [1]. In 2017, the volume of cryptocurrency transactions increased dramatically due to the capital market's ultraexponential growth [2]. However, the movement of cryptocurrency exhibits high volatility, adding more uncertainty to the transaction market. Most articles on cryptocurrency and machine learning focus on the problems of model prediction [3, 4]. However, many ignore the mathematical principles behind the model regardless of the relationship between accuracy and parameter settings. This leads to some seemingly accurate models that are not generally practical. This article will explore the relationship between mathematical principles and model accuracy and discuss the essence through phenomena. Considering that there are two theories in the financial market, one is that the stock price is predictable [5], and the other is that the stock price is entirely unpredictable [6], which indicates that the

price is a Random Walk, the machine learning model described below (e.g., LSTM (Long Short-Term Memory networks) and RNN (recurrent neural network)) will verify the predictable hypotheses, and Random Walk theory is also applied in this article, which will be researched based on the previous study [7–9]. With the global epidemic outbreaks, financial development is primarily affected by COVID-19. The motivation of the study is to explore the machine learning model's performance in both contexts and find an optimal potential parameter combination to explain the unstable trends and some common ones. The study will first experiment with the mentioned financial problem and then apply the result in the COVID-19 model prediction case to verify the model parameters in different contexts and conclude an optimal parameter settings combination.

The Long Short-Term Memory (LSTM) and recurrent neural network (RNN) models are frequently applied in this field, which are preferred over the conventional multilayer perceptron [10]. Sean McNally compared the RNN and the LSTM model used on Bitcoin [11], for the RNN (recurrent

neural network) implementation part. The author first took the temporal length window by the autocorrelation function. In the LSTM part, the previous research [12] has illustrated that, compared with the RNN, the LSTM outperforms RNN and ARIMA at learning long-term dependencies. The ARIMA (Autoregressive Integrated Moving Average) model is a time series model often used in the price prediction [13, 14]. A model comparison is presented [12] in Table 1.

The Long Short-Term Memory (LSTM) and recurrent neural network (RNN) are frequently applied in this field [10]. Table 1 shows that the precision and accuracy do not significantly differ between the two models. Both LSTM and RNN models are capable of training data with LSTM being more applicable to the long-term dependencies.

As for the multiple window length settings [15], different window sizes are applied based on the LSTM model to capture better features of the equipment, which concluded that various time window sizes have a positive impact on recognizing various temporal dependencies among features, while [16] used ten combinations of sliding windows with prediction ranges to explore the accuracy improvement possibility for deep learning fully and concluded that if the window length is small while the prediction range is far ahead simultaneously, the RMSE (Root Mean Square Error) will become lower than the primary method.

A nonlinear model should be applied to this topic in accordance with volatility. Many scholars have compared the RNN (recurrent neural network) and the LSTM (Long Short-Term Memory). According to the results, the LSTM model outperforms RNN, since it is more suitable for long dependencies. Significantly, the window sliding method with the different prediction range variables should be applied in this article. Furthermore, the theory of Random Walks in cryptocurrency prices was also experimented with respect to the predictable price hypothesis.

To sum up, the structure of this article is divided into seven parts: introduction, literature review, methodology, data collection, implementation, discussion, and conclusion and future work. The details are shown in Table 2.

## 2. Literature Review of Related Work

As people's health awareness and philosophy increase, how to more effectively improve the utilization of medical resources has become an issue of concern to society at large. Some literature on medical applications pays particular attention to the wireless sensor network (WSN) technology, a spatially distributed sensor node that aims for important information collection [17]. For example, the study [17] proposed a multiagent-based architecture for WSNs and particle swarm optimization (PSO) algorithm to improve the model ability of the population diversity issue. Besides, based on Ant Colony Optimization (ACO), the study [18] also proposed a novel adaptive intelligent routing scheme for WSNs to achieve a better model performance in terms of energy consumption and efficiency. Moreover, an energy-efficient sleep scheduling mechanism (ESSM) is also proposed for WSNs to reduce energy consumption

effectively [19]. Apart from WSN technology application, deep learning methods and biometric methods are also carried out in some medical issues. For example, the study in [20] used a biometric method, which is a finger vein personal authentication method, and the study in [21] used a deep learning method of XGBoost and genetic algorithm to extract pedestrian feature which is an inspiration of object recognition in medical problems.

## 3. Methodology

### 3.1. Principle and Introduction of LSTM Model

*3.1.1. Start from RNN.* RNN represents the recurrent neural network, and time is a significant impact factor for RNN [22]. The output comes out with each moment's input combined with the state of the current model. In Figure 1, the output  $h_t$  comes out with both the input  $xt$  and the hidden state from moment  $t-1$ , which is provided by the looped edge. Theoretically, the recurrent neural network can be capable of sequences of arbitrary length [23]. However, in practice, the problem of gradient dissipation or explosion will happen during the optimization for the too long sequence. Furthermore, the dissipation of the gradient will make the weight of previous layer not updated during the forward propagation; on the contrary, the gradient explosion will make training process unstable; thus, the model cannot obtain the optimal parameters.

*3.1.2. Mathematical Explanation of RNN.* Given the 3 moments of RNN unit, in Figure 2, assuming that the left input  $S_0$  is a given value and no activation function exists in the neuron, subsequently, the forward process is expressed as

$$\begin{aligned} S_1 &= W_x X_1 + W_s S_0 + b_1 O_1 = W_o S_1 + b_2, \\ S_2 &= W_x X_2 + W_s S_1 + b_1 O_2 = W_o S_2 + b_2, \\ S_3 &= W_x X_3 + W_s S_2 + b_1 O_3 = W_o S_3 + b_2. \end{aligned} \quad (1)$$

At the time of  $t=3$ , the loss function can be written as

$$L_3 = \frac{1}{2}(Y_3 - O_3)^2. \quad (2)$$

RNN training is virtually to seek partial derivatives of  $W_0, W_x, W_s, b_1, b_2$ , adjusting them in order to obtain the minimum of  $L_3$ . According to the chain rule,

$$\begin{aligned} \frac{\delta L_3}{\delta W_0} &= \frac{\delta L_3}{\delta O_3} \frac{\delta O_3}{\delta W_0}, \\ \frac{\delta L_3}{\delta W_x} &= \frac{\delta L_3}{\delta O_3} \frac{\delta O_3}{\delta S_3} \frac{\delta S_3}{\delta W_x} + \frac{\delta L_3}{\delta O_3} \frac{\delta O_3}{\delta S_3} \frac{\delta S_3}{\delta S_2} \frac{\delta S_2}{\delta W_x} + \frac{\delta L_3}{\delta O_3} \frac{\delta O_3}{\delta S_3} \frac{\delta S_3}{\delta S_2} \frac{\delta S_2}{\delta S_1} \frac{\delta S_1}{\delta W_x}, \\ \frac{\delta L_3}{\delta W_s} &= \frac{\delta L_3}{\delta O_3} \frac{\delta O_3}{\delta S_3} \frac{\delta S_3}{\delta W_s} + \frac{\delta L_3}{\delta O_3} \frac{\delta O_3}{\delta S_3} \frac{\delta S_3}{\delta S_2} \frac{\delta S_2}{\delta W_s} + \frac{\delta L_3}{\delta O_3} \frac{\delta O_3}{\delta S_3} \frac{\delta S_3}{\delta S_2} \frac{\delta S_2}{\delta S_1} \frac{\delta S_1}{\delta W_s}. \end{aligned} \quad (3)$$

It is briefed as

TABLE 1: The results of different model performance [12].

Model	Temporal length	Sensitivity (%)	Specificity	Precision	Accuracy (%)	RMSE (%)
LSTM	100	37	61.30%	35.50%	52.78	6.87
RNN	20	40.40	56.65%	39.08%	50.25	5.45
ARIMA	170	14.7	1	1	50.05	53.74

TABLE 2: Structure details of the article.

Section	Objective
Introduction	To clarify the research background and deep learning models as well as the article structure.
Literature review	To clarify the related work in the current topic.
Methodology	To introduce the mathematical principle of LSTM and Random Walk model.
Data collection	To introduce the data source.
Implementation	To train the models applied in financial and medical issues.
Discussion	To discuss the LSTM model performance and parameter settings in financial and medical cases.
Conclusion and future work	To give the final conclusion and future work suggestions.

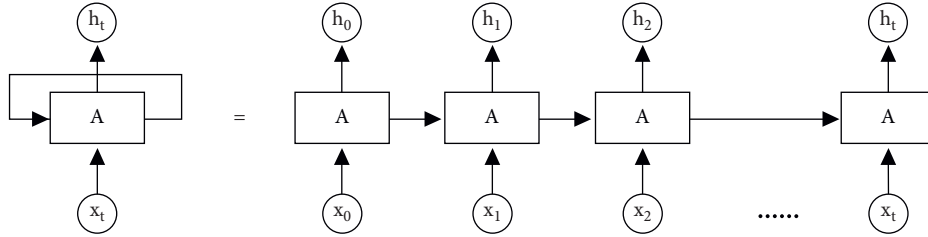


FIGURE 1: The structure of RNN.

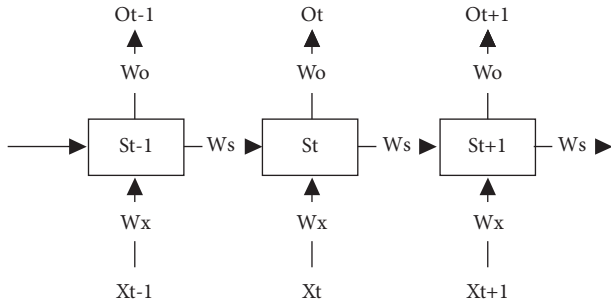


FIGURE 2: The inner structure of RNN.

$$\frac{\delta L_3}{\delta W_x} = \sum_{k=0}^t \frac{\delta L_t}{\delta O_t} \left( \prod_{j=k+1}^t \frac{\delta S_j}{\delta S_{j-1}} \right) \frac{\delta S_k}{\delta W_x}, \quad (4)$$

$$\frac{\delta L_3}{\delta W_s} = \sum_{k=0}^t \frac{\delta L_t}{\delta O_t} \frac{\delta O_t}{\delta S_t} \left( \prod_{j=k+1}^t \frac{\delta S_j}{\delta S_{j-1}} \right) \frac{\delta S_k}{\delta W_s}.$$

This formula suggests that the  $\prod_{j=k+1}^t \delta S_j / \delta S_{j-1}$  part causes the gradient dissipation or explosion. With the activation function added, it is expressed as

$$S_j = \tanh(W_x X_j + W_x S_{j-1} + b_1), \quad (5)$$

and it is concluded that

$$\prod_{j=k+1}^t \frac{\delta S_j}{\delta S_{j-1}} = \prod_{j=k+1}^t \tanh' W_s, \quad (6)$$

where  $\tanh$  derivative is always below 1. With the increase in  $t$ , the above formula's value turns closer to zero as long as  $W_s$  is above 0 and below 1 as well, leading to the disappearance of the gradient. Subsequently, the above formula will become more and more infinite if  $W_s$  is large, thus producing a gradient explosion, which explains why the LSTM is introduced.

**3.1.3. Mathematical Explanation of LSTM Model.** LSTM represents the Long Short-Term Memory, an RNN type.  $C_t$  is called current cell state, which can be expressed as

$$c_t = f_t \oplus c_{t-1} + i_t \otimes \tanh(W_c [h_{t-1}, x_t] + b_c), \quad (7)$$

and  $f_t$  is called the forget gate, which can be expressed as

$$f_t = \sigma(W_f [h_{t-1}, x_t] + b_f), \quad (8)$$

deciding which features can be employed for the calculation of  $C_t$  from  $C_{t-1}$ . The current hidden output can be expressed as

$$h_t = o_t \otimes \tanh(c_t). \quad (9)$$

Besides, the input and output gates are expressed, respectively, as



$$\begin{aligned} i_t &= \sigma(W_i[h_{t-1}, x_t] + b_i) \\ o_t &= \sigma(W_o[h_{t-1}, x_t] + b_o). \end{aligned} \quad (10)$$

The above formulas show that the activation function of 3 gates is sigmoid, revealing that the output of these three gates is either close to 0 or close to 1. This makes  $\delta c_t / \delta c_{t-1} = f_t$ ,  $\delta h_t / \delta h_{t-1} = o_t$  part become 0 or 1. When it is 1, the gradient can be transmitted well in the LSTM, significantly reducing the probability of the gradient dissipation. When the gate is 0, the information at the previous moment does not impact the current moment, indicating that there are no instructions to transmit the gradient backwards for updating the parameters [24]. Accordingly, this explains the reason why the gradient can be solved using the LSTM model shows in Figure 3.

*3.1.4. Mathematical Explanation of Random Walk Model.* For the time series  $\{x_t\}$ , if it satisfies  $x_t = x_{t-1} + w_t$ , where  $w_t$  denotes a white noise with a mean of 0 and a variance of  $\sigma^2$ , the sequence  $\{x_t\}$  will be a Random Walk [37]. By definition,  $t$  at any  $x_t$  moment refers to the sum of all historical white noise sequences that do not exceed the  $t$  moment, so it is concluded that

$$x_t = w_t + w_{t-1} + w_{t-2} + \dots + w_0. \quad (11)$$

The sequence mean and variance of Random Walk are presented as follows:

$$\begin{aligned} \mu_{x_t} &= 0, \\ \text{var}(x_t) &= \text{var}(w_t) + \text{var}(w_{t-1}) + \dots + \text{var}(w_0) \\ &= t \times \text{var}(w_t) \\ &= t\sigma^2. \end{aligned} \quad (12)$$

Although the mean does not change with time  $t$ , due to the fact that the variance is the function that relates to  $t$ , the Random Walk does not satisfy the stability. As time  $t$  and the variance of  $x_t$  are regulated, the stability is upregulated. For the given interval  $k$ , the Random Walk covariance is performed as

$$\begin{aligned} \text{Cov}(x_t, x_{t+k}) &= \text{Cov}(x_t, x_t + w_{t+1} + \dots + w_k) \\ &= \text{Cov}\left(x_t, x_t + \sum_{i=t+1}^k \text{Cov}(x_t, w_i)\right) \\ \text{Cov} &= (x_t, x_t) + 0 \\ &= t\sigma^2. \end{aligned} \quad (13)$$

From the concluded variance and covariance, the autocorrelation function  $\rho_k(t)$  is calculated as follows:

$$\begin{aligned} \rho_k(t) &= \frac{\text{Cov}(x_t, x_{t+k})}{\sqrt{\text{Var}(x_t)}\sqrt{\text{Var}(x_{t+k})}} \\ &= \frac{t\sigma^2}{\sqrt{t\sigma^2}\sqrt{(t+k)\sigma^2}} \\ &= \frac{1}{\sqrt{1+k/t}}. \end{aligned} \quad (14)$$

Clearly, the autocorrelation function is related to time  $t$  and interval  $k$ , indicating that if the Random Walk model has a long time series while the interval is quite small, the autocorrelation coefficient is approximated as 1. In other words, if there is a model predicting the stock price based on time  $t$  as the forecast for the  $t+1$  value, the correlation coefficient between the actual value and the predicted value equals the stock price sequence of  $k=1$ . In other words, the forecast of today's price as tomorrow's price is also very close to 1, which will mislead us into thinking that the model is accurate.

## 4. Data Collection

The financial data are all collected from the CoinMarketCap, which is an authoritative website committed to cryptocurrency market value statistics. Only the Bitcoin and Ethereum data are adopted to train the LSTM model and the Random Walk model. The raw ranges from April 2017 to December 2020 for nearly 3 years span. The training size parameter is 0.8, while the test size reaches 0.2. Meanwhile, the COVID-19 cases data is obtained in National Statistical Office, ranging from March 2020 to July 2020 in China; given that the mentioned period witnessed the peak of the global epidemic, it might be representative.

## 5. Implementation

### 5.1. Training Process of Random Walk Model

*5.1.1. Single-Point Method Prediction.* From the preliminaries illustrated below, the Random Walk model will learn parameter  $\sigma$ , which is the only parameter of the Random Walk. Figure 4 shows the model performance.

Based on the preliminaries, the single-point Random Walk model seems to be performing well, which is in accordance with expectation. The model just predicts the next day, so  $k=1$ . Besides, the time span is 3 years, suggesting that  $t$  is very large, so  $\rho, k=1$ , implying that the forecast of next day is just the repeat of the current day, and, due to the single-point method selection, the error will reset every time, which means that every next input will be the true data. Figures 5 and 6 suggest that the prediction line is similar to the copy in the horizontal direction. The model seeming accurate is attributed to the mathematical nature of Random Walk rather than the training process. Here, the model trained by the data in 2017 shows the details of the copy in the horizontal direction.

*5.1.2. Multipoint Method Prediction.* As mentioned below, if the model intends to ignore the misleading accuracy caused by the nature of Random Walk [25], increasing the value of  $k$  can solve this problem. That is to say, the interval of the Random Walk step will be larger instead of +1 days. Therefore, a multipoint prediction method is proposed. In such way, the error cannot be reset, which will be exacerbated by subsequent predictions. The training result can be seen in Figure 7.

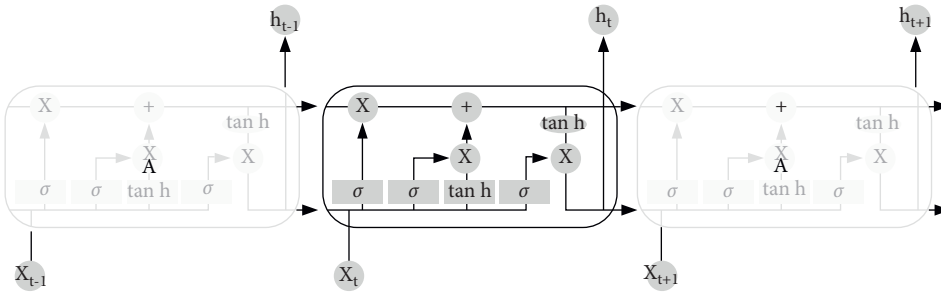


FIGURE 3: The structure of LSTM.

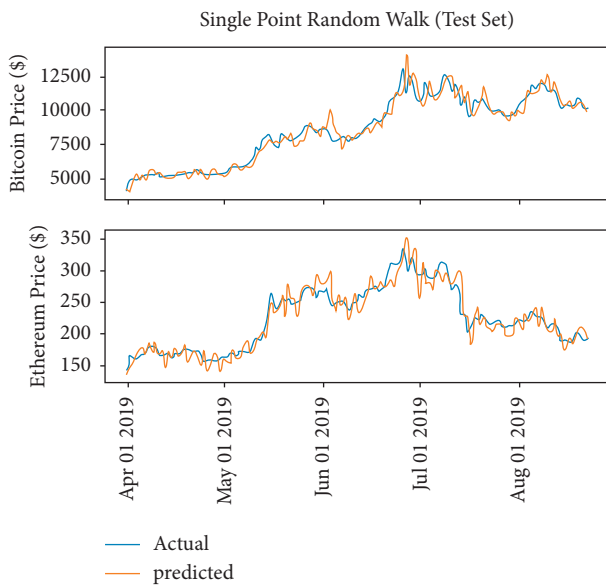


FIGURE 4: Single point Random Walk model performance.

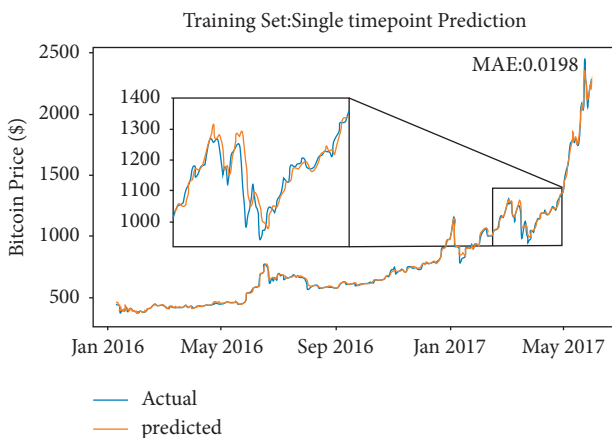


FIGURE 5: The details of single-point prediction on Bitcoin.

Obviously, changing the value of  $k$  will cause a significant reduction in the model accuracy;  $p_k(t)$  will not approach 1 with the increase of  $k$ . That is to say, the result of the model is not associated with the nature of the Random Walk

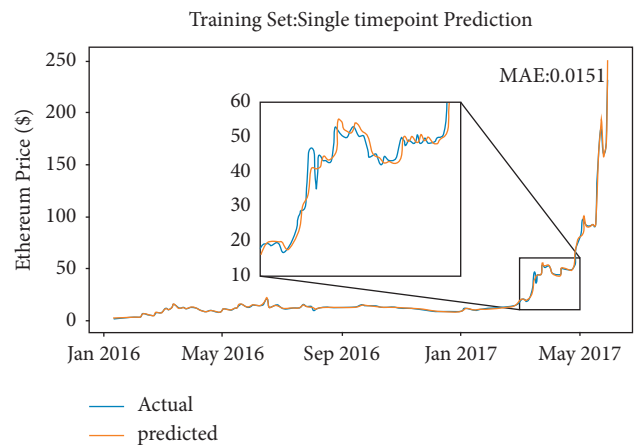


FIGURE 6: The details of single-point prediction on Ethereum.

model. What is more, because the errors will be compounded by subsequent predictions, the predicting line is penalized seriously. What needs to be noticed is the fact that the Random Walk model is defined as  $x_t = x_{t-1} + w_t$ . That is, the price of the day is randomly changed based on the price of the previous day, while the price difference is all included in the random item  $w_t$ . It can be seen from the above Random Walk model that the time series of the securities price will be in a random state and will not exhibit a certain observable or statistically determined trend. Compared with the machine learning model, the Random Walk model only explores the random item  $w_t$ ; it does not learn from the inputs or learn any parameters or weights of the model. That is why the single-point model or the multipoint model are both not the ideal solution for predicting the trend of cryptocurrency.

### 5.2. Training Process of LSTM Model

**5.2.1. Point-to-Point Method Prediction.** The LSTM created is a two-dimensional model using only the close price and the transaction volume features, considering the price of changes daily is an immense difference every period as Figure 8 shows, which means that the model will not converge, so the normalizing operation [26] might be required.

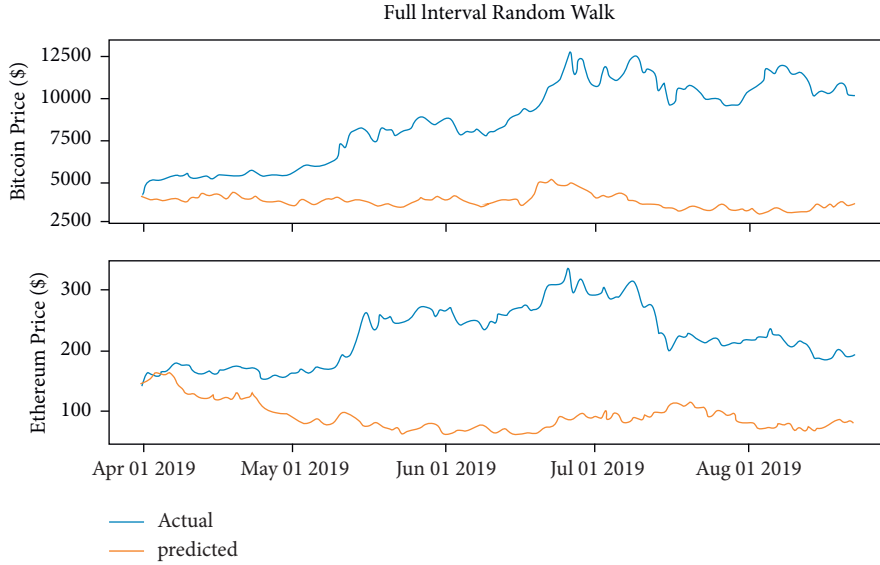


FIGURE 7: Full interval Random Walk performance.

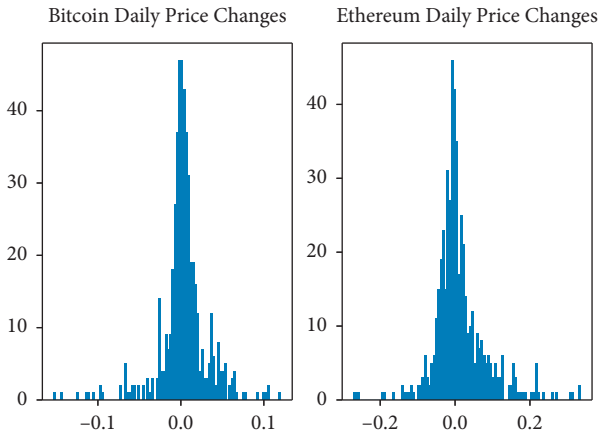


FIGURE 8: Daily price changes on Bitcoin and Ethereum.

For the training data, to normalize the price changes, equation (15) is used, where  $p_i$  represents the current window price and  $p_0$  is the next window price. So the input and output will be a percentage format. For the test data, the output will be denormalized as a direct real price of prediction is expected to visualize; for the denormalization, equation (16) will be used.

$$n_i = \left( \frac{p_i}{p_0} - 1 \right), \quad (15)$$

$$p_i = p_0(n_i + 1). \quad (16)$$

Here the model uses MAE (Mean Absolute Error) [27] equation to validate the error between the predicted value and the true value, which is the average of absolute errors that can better reflect the actual situation of the prediction value error.

TABLE 3: LSTM single-point prediction training process.

Epoch	18/20	19/20	20/20
step_loss	0.031	0.029	0.031
mean_absolute_error	0.031	0.029	0.031
val_loss	0.023	0.024	0.023
val_mean_absolute_error	0.023	0.024	0.023

After the selection of parameters, the training dataset is used to train the model. The merge date starts from 2017 to 2020 and the split size is 0.8, so the training dataset is mainly from 05 in 2017 to 10 in 2019. Table 3 shows the Bitcoin training process of the model; it is obvious that ,from epoch 18, the model started to converge as it lastly nearly stays at the MAE of 0.0330. Figure 9 shows the LSTM training process of the Bitcoin.

After the convergence of the model [28], it is applied to the test dataset, which ranges from 11 in 2019 to 12 in 2020. The performance of the model on the Bitcoin test dataset is shown in Figure 10. As in Figure 11, both training set and the test set stop decreasing at epoch 20; after epoch 20, the training set error will still decrease, but the error on the test set will start to increase due to the model overfitting problem. Figures 12 and 13 show the performance of the model on Ethereum.

The model in this part used the point-to-point method. The point-to-point prediction is the process of making the model predict one single-point value each time and plot the corresponding position in the figure; after predicting this point, the window will slide to next point with the complete test data. Besides, the point-to-point method seems to be more accurate than the full interval prediction [29], whereas it does not imply that the point-to-point model outperforms the full interval model, since the error generated by each single prediction is reset each time, the neural network itself



FIGURE 9: The explanation of window length and prediction range.

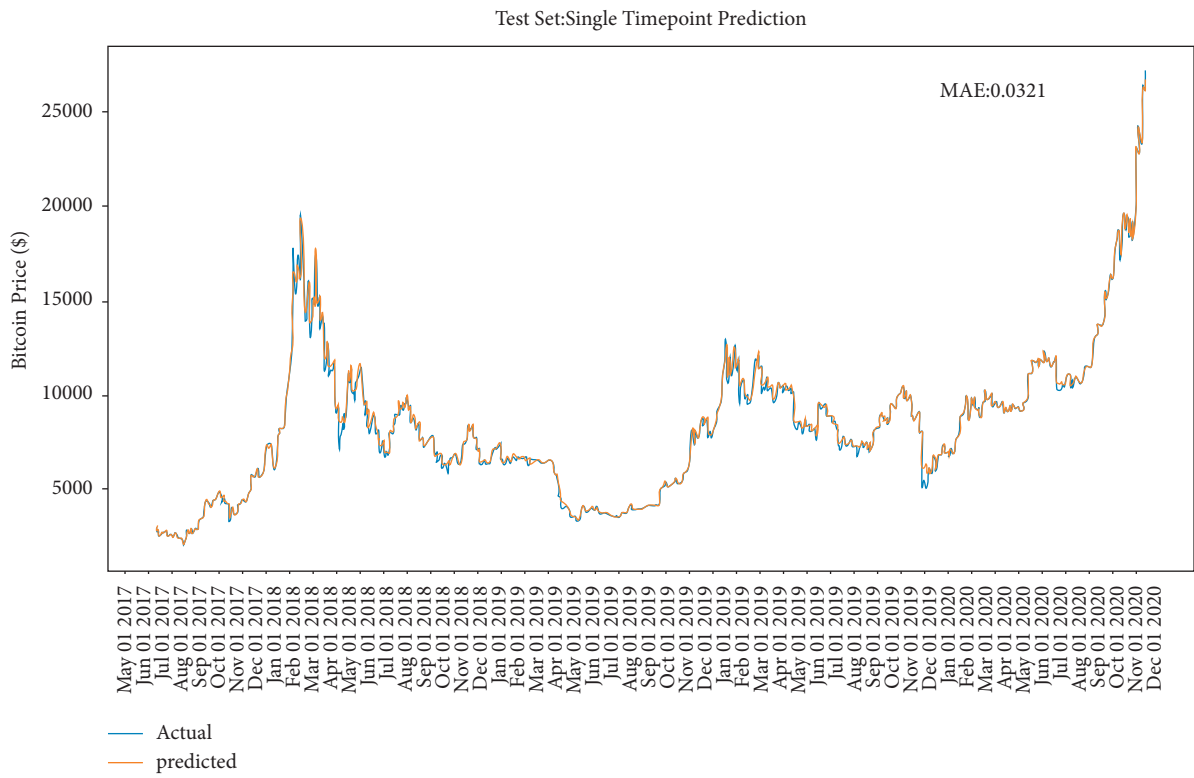


FIGURE 10: The performance of LSTM with single-point prediction on Bitcoin.

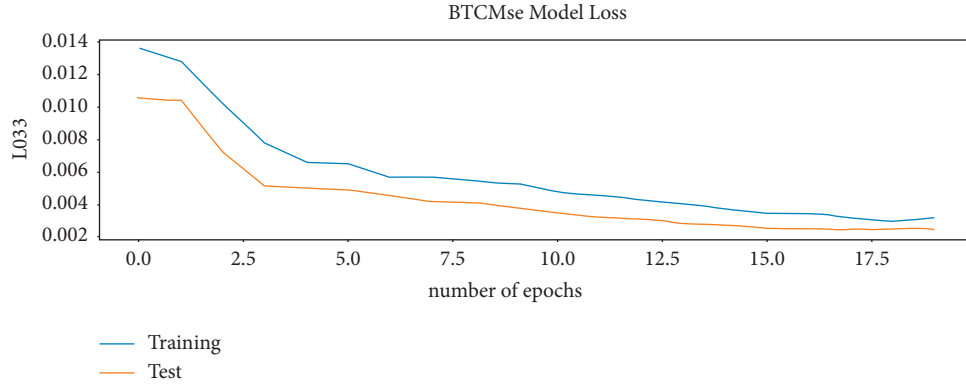


FIGURE 11: The model loss on Bitcoin.

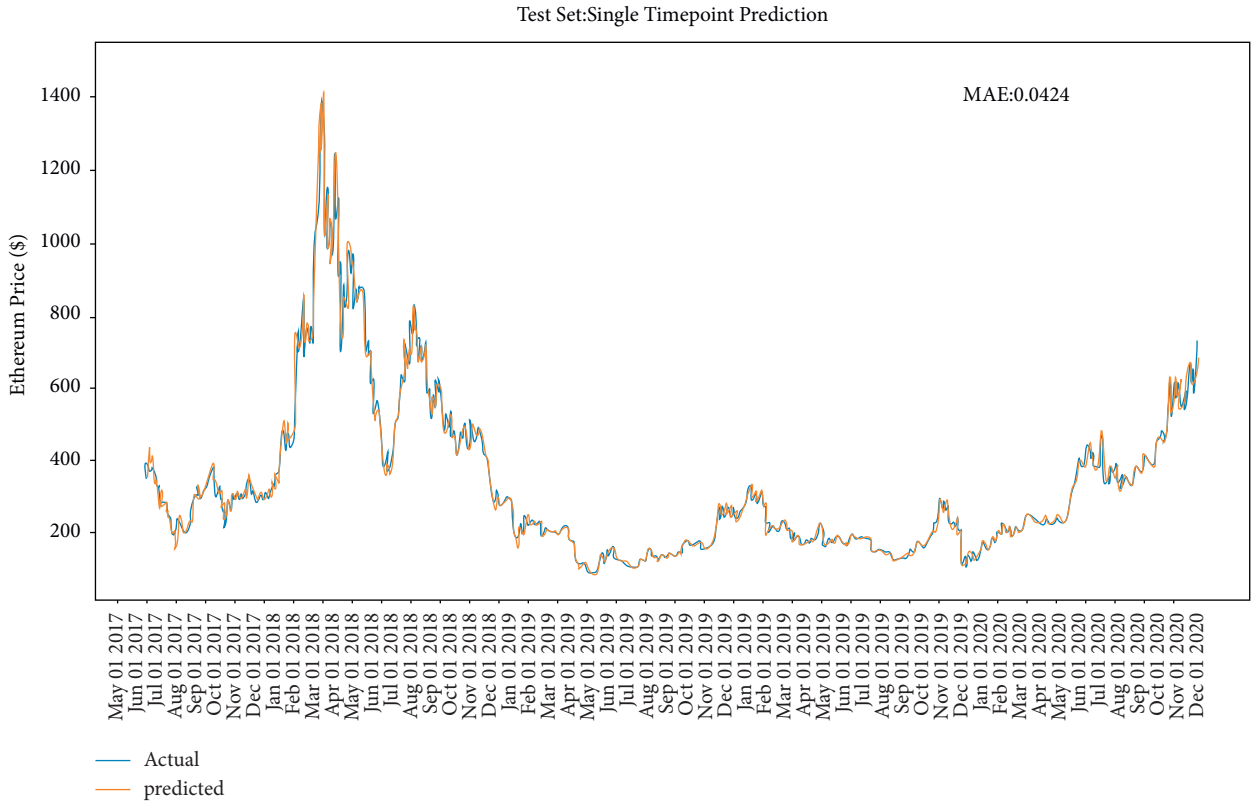


FIGURE 12: The performance of LSTM with single-point prediction on Ethereum.

does not need to know the time series itself, and all the inputs are based on the real value in every next prediction. For the ignorance of the errors, the model seems unsurprisingly accurate. Furthermore, in Figure 14, it is suggested that the predicted value is more like a horizontal translation of the true value. For instance, from mid-May to mid-June 2019, several prices increased, and the peaks were following the fluctuations of the true values, which has an obvious hysteresis. In other words, the deep learning LSTM model regenerates an autoregressive model of order  $p$ ; in these

datasets area, the predicted value is the weighted sum of the previous  $p$  values, as defined below:

$$\text{PredPrice} = w_0 + w_1 * \text{Price}_{t-1} + \dots + w_p * \text{Price}_{t-p} + \epsilon_t, \epsilon_t \sim N(0, \sigma), \tag{17}$$

where the next prediction will only be the true  $\text{Price}_{t-p}$  value with the calculated weight because the point-to-point method will ignore the error of every previous prediction, which largely reduces the inaccuracy. Therefore, in order to

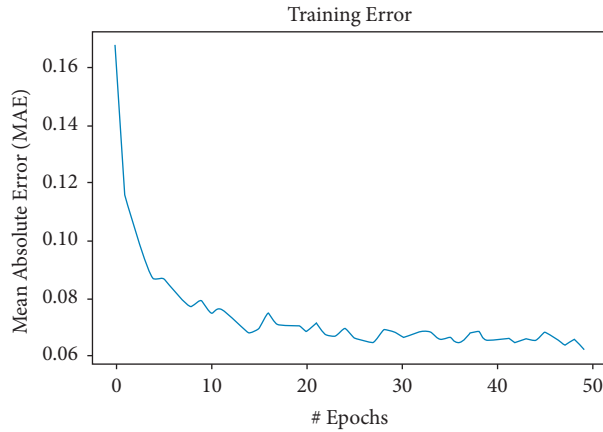


FIGURE 13: The training error on Ethereum.

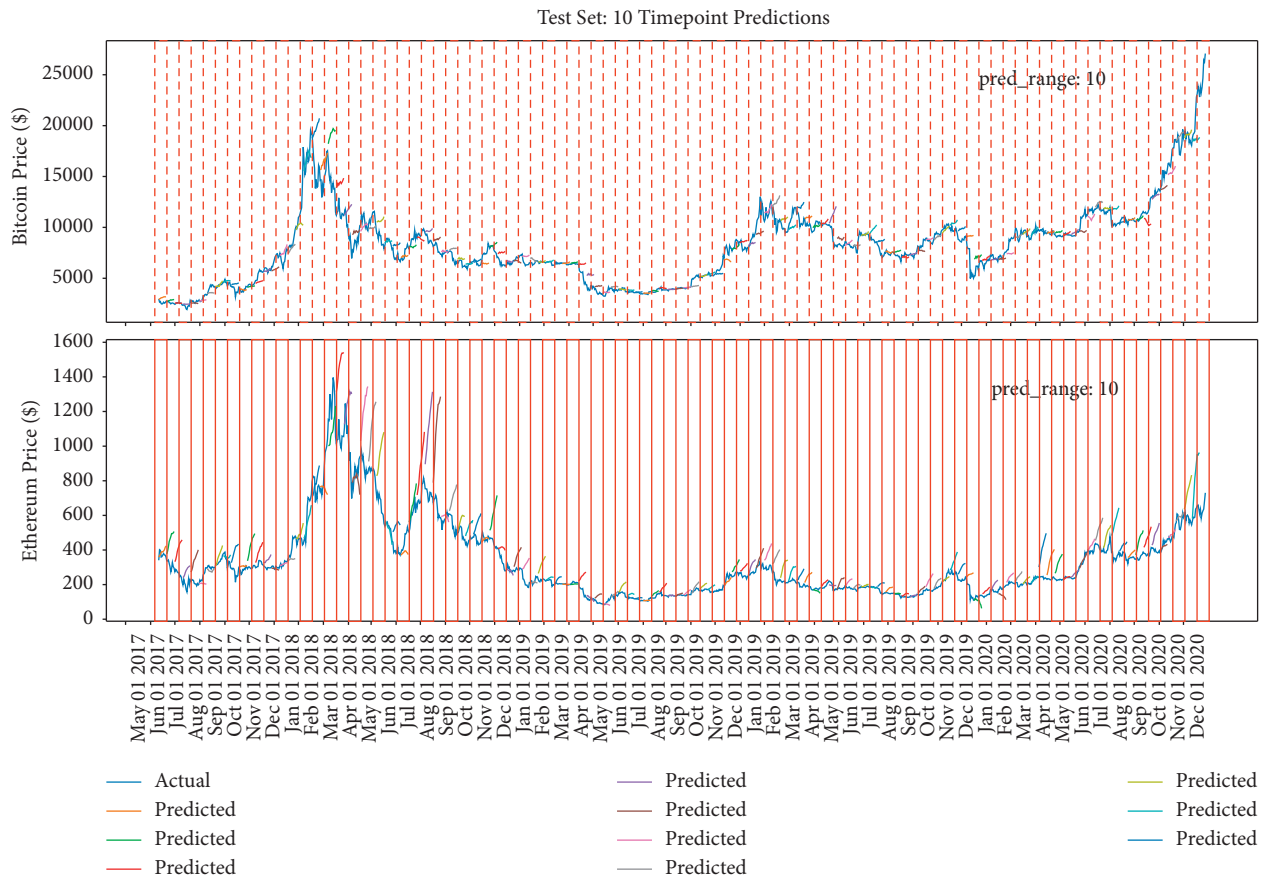


FIGURE 14: The model's performance with window length = 10 and prediction range = 10.

TABLE 4: The error subtraction based on single day.

	Window length = 10		
Days interval (based on single day)	4	9	14
Error subtraction, Bitcoin	0.005	0.013	0.025
Error subtraction, Ethereum	0.068	0.103	0.121

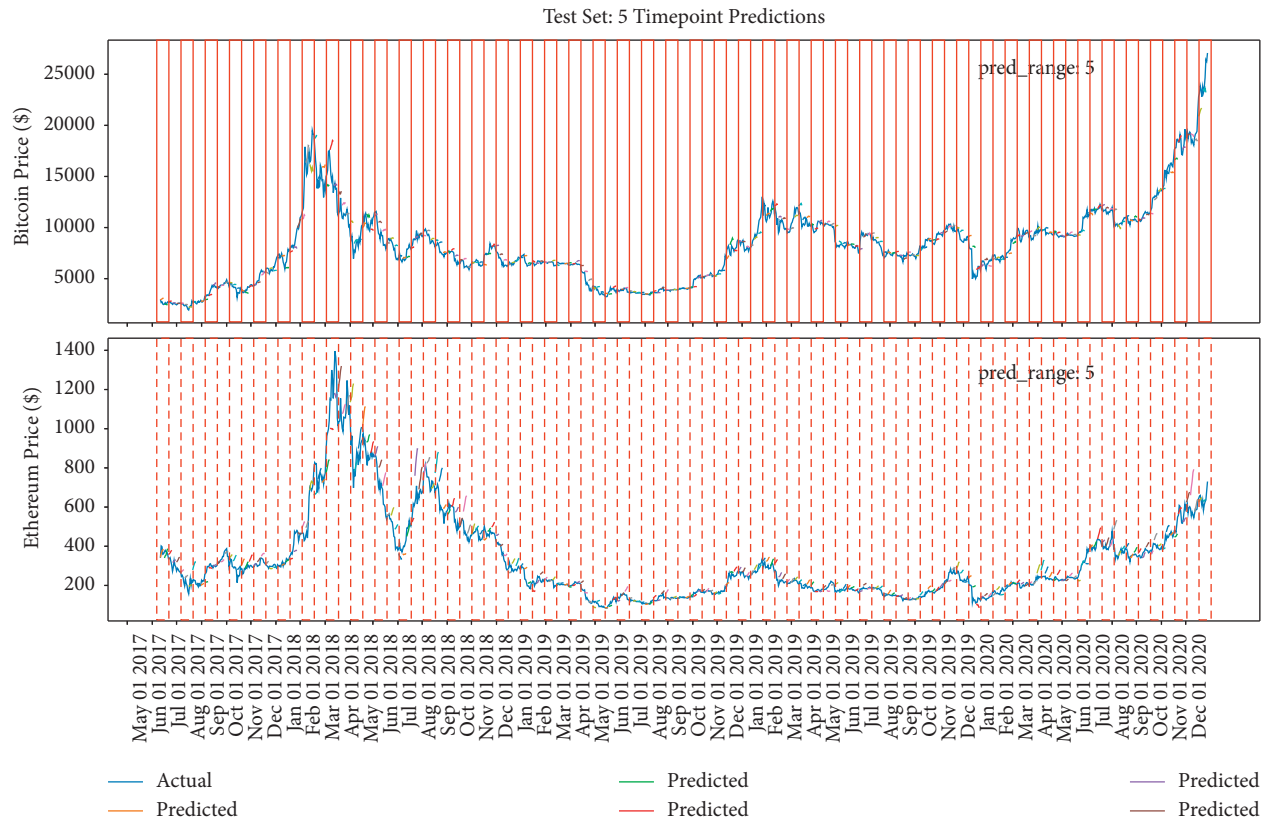


FIGURE 15: The model's performance with window length = 10 and prediction range = 5.

maximize the advantages of LSTM based on time series and avoid the model updating the error at every step, the model will be improved from the two following indicators: the first is window length, and the second is the prediction range. Window length is the historical time used by LSTM, and the prediction range refers to the range of backward prediction by the data trained in window length (Figure 9).

**5.2.2. Multipoint Method Prediction with Fixed Window Length Selection.** Unlike the limitation of point-to-point training method, the multiple time-point method is more practical. Likewise, it initializes the test window and keeps moving to predict next point [30]. Besides, it will move forward a full window size and resets the window with true test data, while it moves to the point where the input window is already constituted by full past  $X_{t-1}$  predictions. Thus, during the prediction, the error will not be fully reset, whereas the error will be accumulated in each full predicted window length, and the error will be reset again in a new window length. The error subtraction is shown in Table 4. For this reason, the model will be more practical. It is neither as deceptive as single-point prediction nor does it completely detour the model from the trajectory of the real point.

Figures 15, 14, and 16 show that the multiple sequence LSTM does not perform well as expected. Besides, the red line in the figure is the prediction range. In the training process, the prediction range is set, respectively, at

[5, 10, 15], while the window length is set at 10. The prediction of the model in each range does not reflect the price of the next trend, and the model seems to only predict the upward trend of the trend, while the price decline trend model does not seem to be aware. This may be due to the selection of parameters or the selection of the length of the window, which reduces the model accuracy. In addition, Figure 17 points out that when the window length is fixed, as the number of prediction points increases, the MAE increases accordingly, which indicates that, in the condition of the same window length selection, the model will be more accurate with less number of points.

**5.2.3. Fixed Multipoint Method Prediction with Different Window Length Selection.** The previous part verifies the impact of different amount of points selection on the model when the window length is fixed. This part will verify the impact of different window length on the accuracy of the model when the amount of points selection is fixed. Similarly, the red lines in Figures 18–20 refer to the different window length settings. In the training process, the window length is set, respectively, at 10, 50, and 90, while the prediction range is set at 5. Figure 18 shows that, at the condition of window length = 10, the model prediction trend performs similarly to the previous part, which seems to only predict the upward trend regardless of the decrease trend, while when the window length = 50, the model could reflect the correct decrease trend

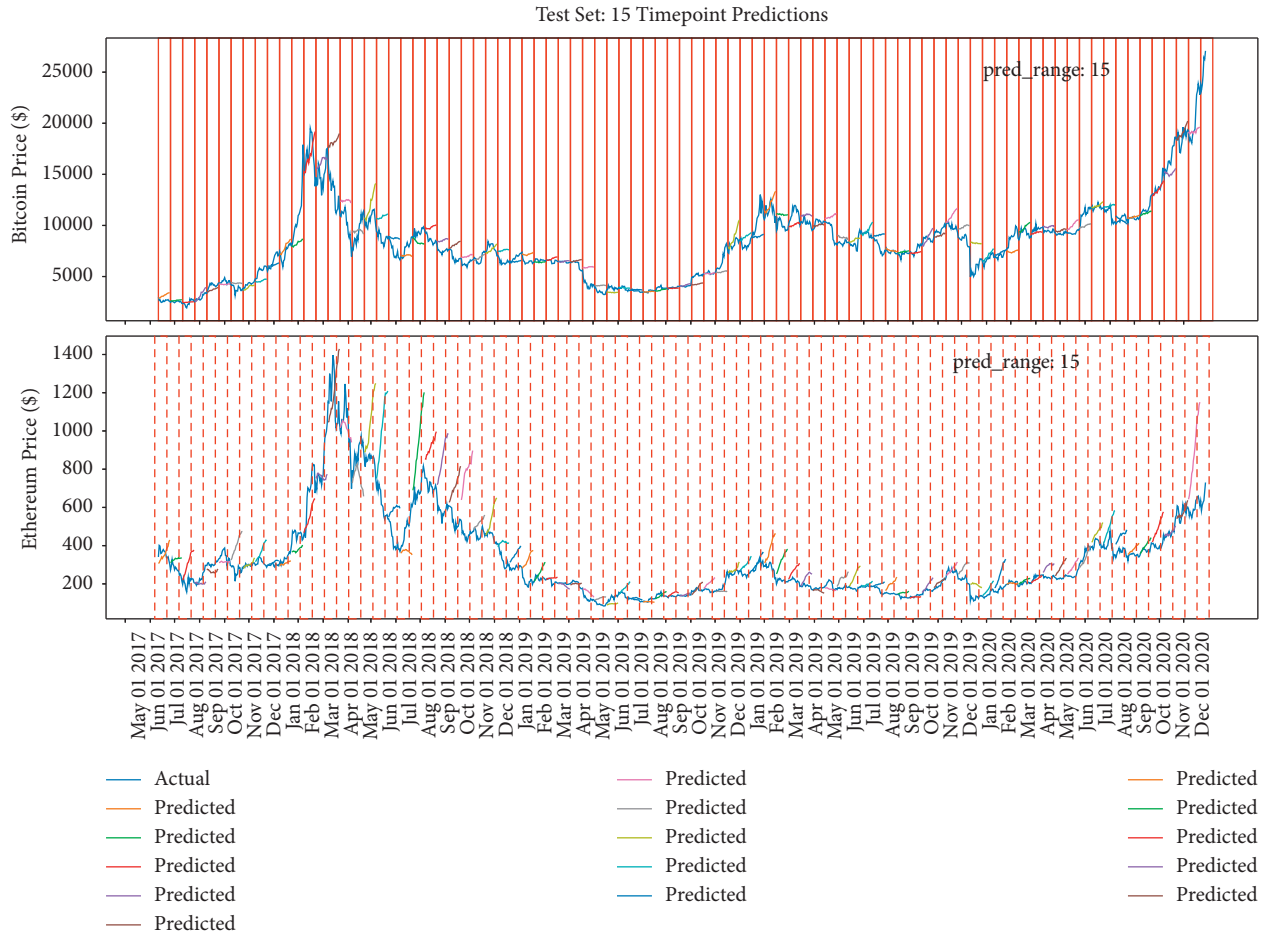


FIGURE 16: The model’s performance with window length = 10 and prediction range = 15.

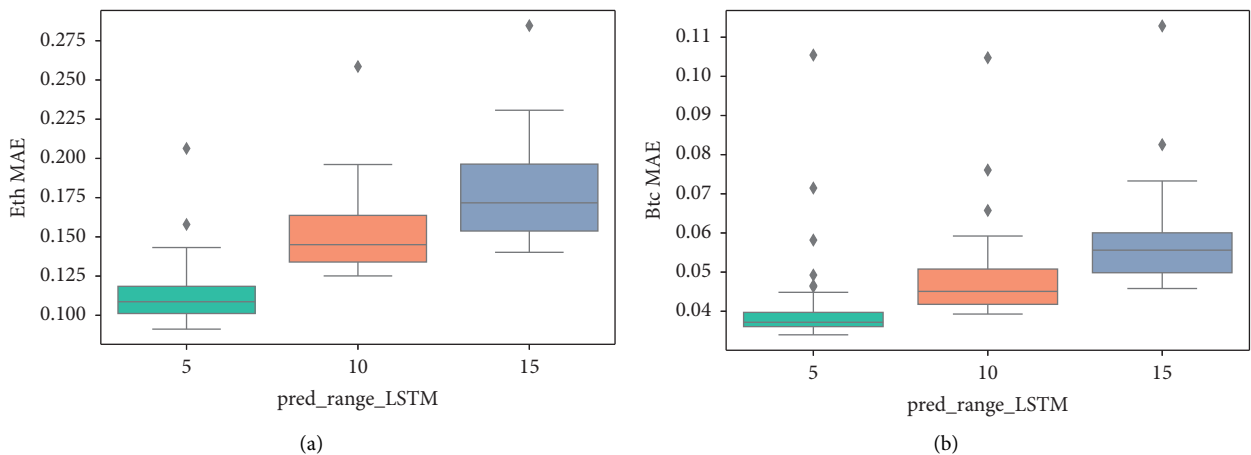


FIGURE 17: The box-plot of MAE based on different prediction ranges. (a) Ethereum MAE. (b) Bitcoin MAE.

generally, and when the window length = 90, the model could reflect all the trend but is not basically right; especially during the period from May 2019 to August 2019, the decrease trend of Bitcoin prediction is totally wrong.

From Figure 21, it could be concluded that, with the fixed prediction range, the model accuracy decreases with the increase of the window length, which is caused by the accumulation of errors [31] in the model.



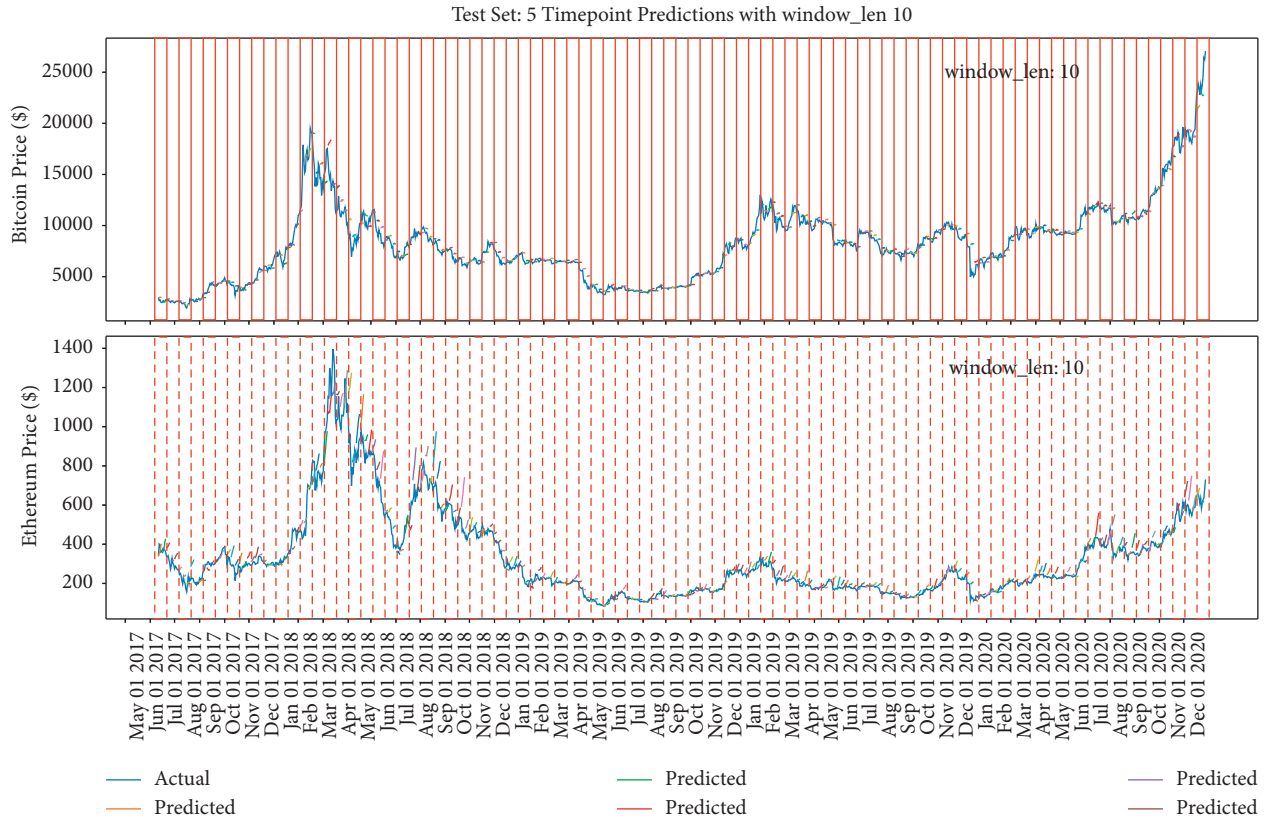


FIGURE 18: The model's performance with window length = 10 and prediction range = 5.

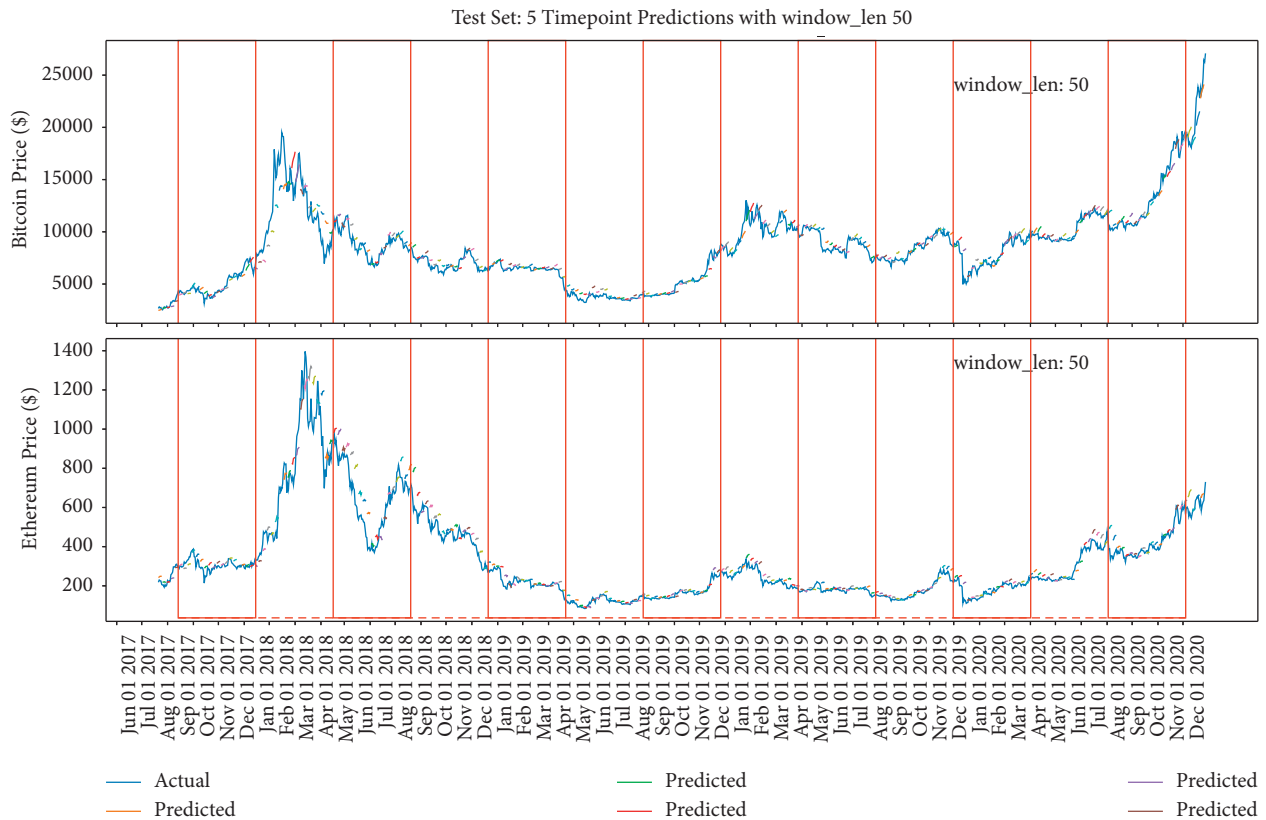


FIGURE 19: The model's performance with window length = 50 and prediction range = 5.

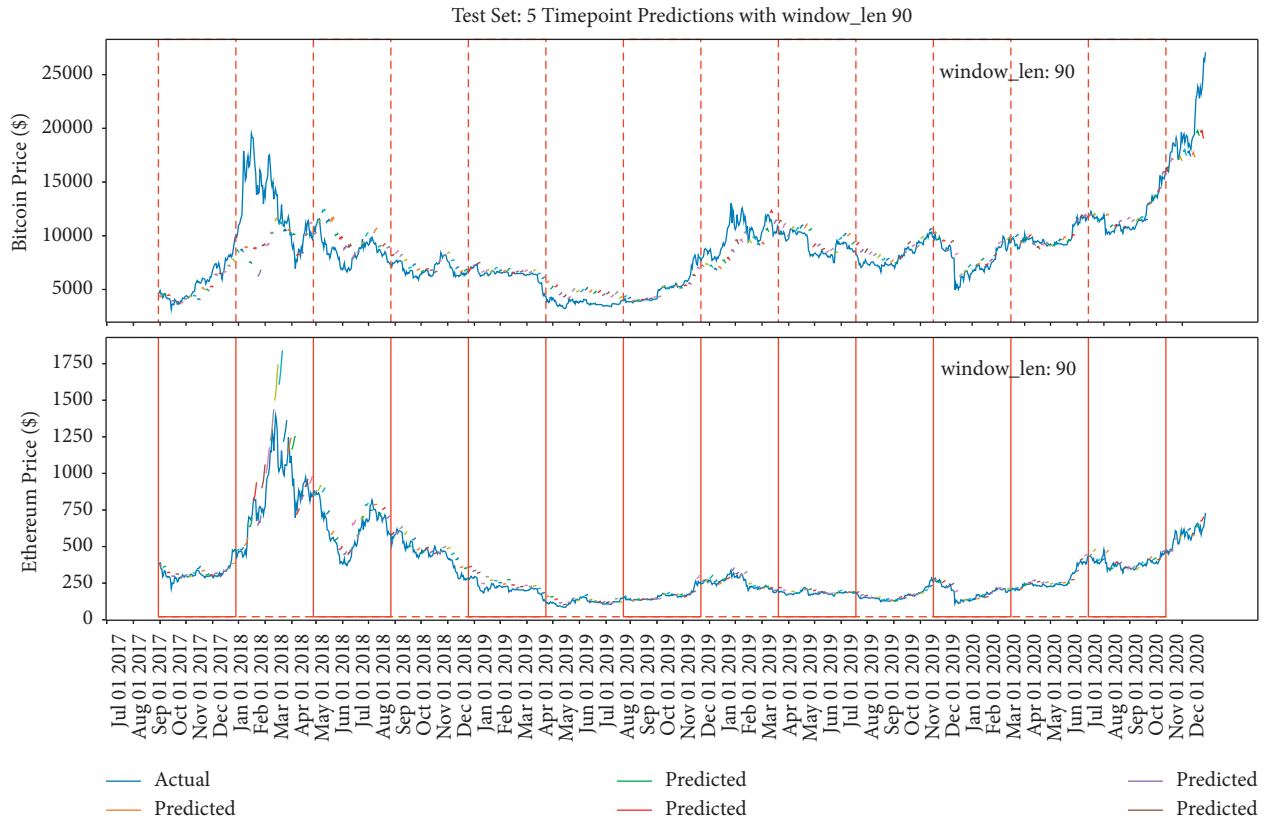


FIGURE 20: The model’s performance with window length = 90 and prediction range = 5.

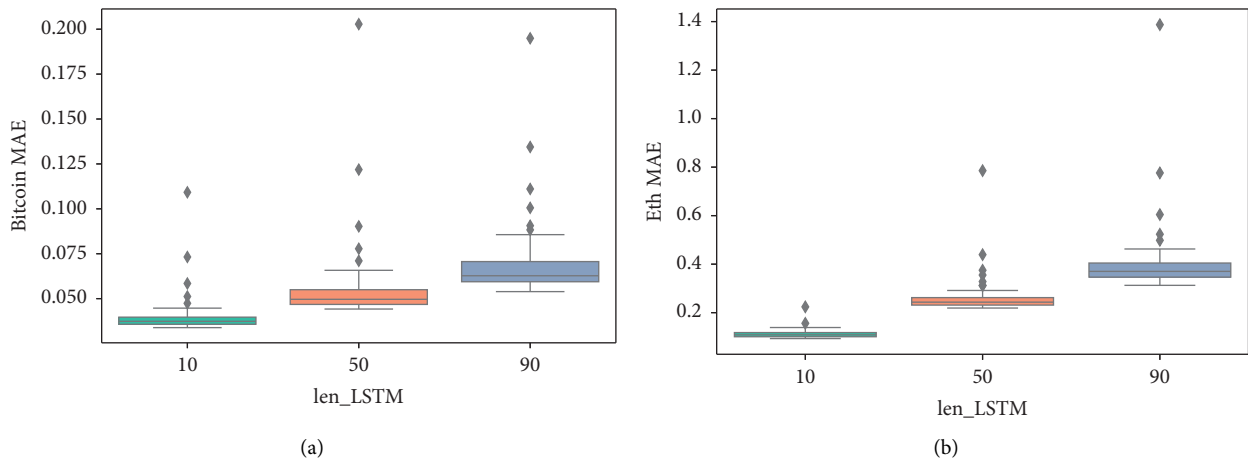


FIGURE 21: The box-plot of MAE based on different window length. (a) Bitcoin MAE. (b) Ethereum MAE.

5.2.4. *Exploration of LSTM Performance on COVID-19.* According to the illustration of prediction range and window length based on the LSTM regarding the cryptocurrency problems, the parameters are concluded as 5 and 10, respectively. In order to test if other circumstances also satisfy this parameter combination, a COVID-19 growth cases per day prediction is introduced. It could be seen that the epidemic trend decreased significantly from the start of

March to mid-March and increased back to 150 cases per day. The diagnosed cases reached the peak and decreased dramatically in mid-April and gradually maintained a flattening trend. Figure 22 shows the trends.

In order to explore the different parameter settings effect based on the COVID-19 data, three combinations of window length and prediction range are applied, and the results are shown in Figures 23–25. The grey-dotted lines in the

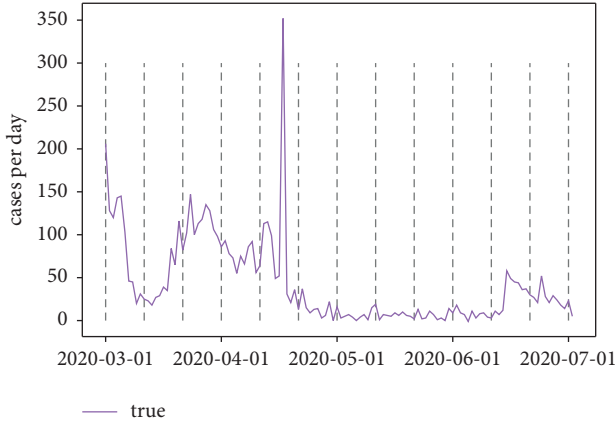


FIGURE 22: COVID-19 trend in 2020 (March to June).

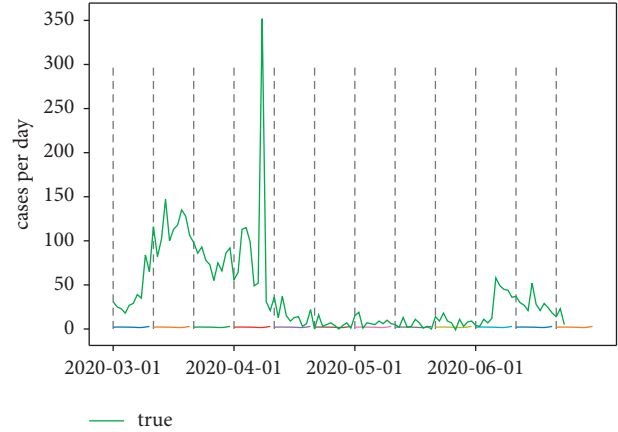


FIGURE 25: Window length = 10; prediction range = 10.

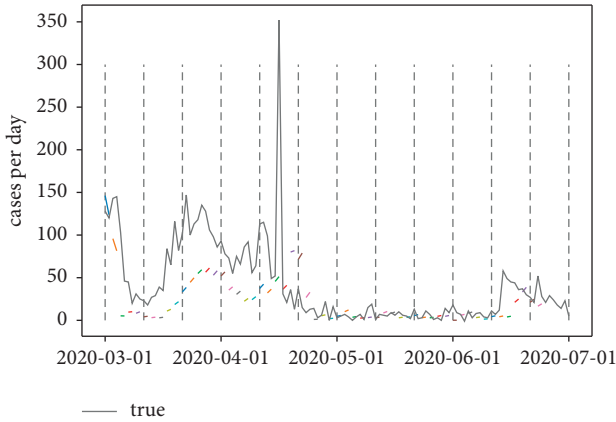


FIGURE 23: Window length = 10; prediction range = 1.

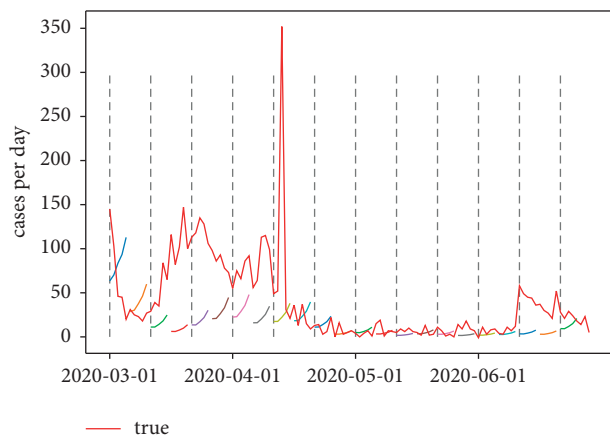


FIGURE 24: Window length = 10; prediction range = 5.

following figures are the certain window length of 10, whereas the short lines represent the different prediction range.

From Table 5, it is clear that the model prediction effect was the same at the parameter's combination of 10 and 5,

TABLE 5: The results of window length = 10 with different prediction range of COVID-19.

	Window length = 10		
Prediction range	1	5	10
Loss error	0.0016	0.0015	0.0016

which is in line with the financial problems mentioned previously. The loss error of the LSTM model reached minimum at 0.0015; what should be noticed is the fact that the peak time during mid-April was not reflected by the model; given that the window length is 10, the model is limited by its deferral nature. In most cases, however, the model can exhibit some degree of slowdown in the rate of growth, which might be suggestive for the medical officials.

## 6. Discussion

According to Tables 3, 5, and 6, after using different combinations of window length and prediction range, it is found that when window length = 10 and prediction range = 5, the Bitcoin and Ethereum LSTM models reach minor errors, which are 0.037 and 0.113, respectively; in the circumstances of predicting the COVID-19 case, the model applied the same parameters combinations in line with the financial problem. It turned out that the parameters window length = 10 and prediction range = 5 are also optimal. In this regard, the combination of window length = 10 and prediction range = 5 should be suggested in future work. Besides, it can be seen that although the single-point method has the smallest error, it is the result of the error being reset every time. However, in the real financial price market, only predicting the price trend of the next day is impractical. Thus, predicting the price over a period of time with a proper error reset frequency is more practical, that is, to have a specific prediction range.

Table 7 shows the relationship between the interval of days and the accuracy. It can be seen that the error is not as significant as expected. Therefore, it can be concluded that, with a particular model accuracy guaranteed, the model has

TABLE 6: The results of fixed prediction range with different window length.

Window length	Prediction range = 5		
	10	50	90
LSTM MAE, Bitcoin	0.037	0.051	0.063
LSTM MAE, Ethereum	0.113	0.312	0.387

TABLE 7: The results of fixed window length with different prediction range.

Prediction range	Window length = 10			
	1 (single point)	5	10	15
LSTM MAE, Bitcoin	0.032	0.037	0.045	0.057
LSTM MAE, Ethereum	0.042	0.113	0.145	0.163

the best balance of practicality and accuracy based on the combination of window length = 10 and prediction range = 5. In summary, it can be concluded from the study of the Random Walk model and LSTM model that it is not appropriate to only focus on the model accuracy; considering the parameter setting and mathematical meaning, as well as practicality, also matters. Therefore, the balance between model practicality and accuracy is crucial.

## 7. Conclusion and Future Work

The study compared the effectiveness of the LSTM and that of the Random Walk model in terms of financial issues. Moreover, the study explored the LSTM algorithm in combination with different parameter settings regarding different circumstances, respectively, the financial and medical issues. The main conclusion is that the LSTM model performs better than the Random Walk model. What needs to be noticed is the fact that the optimized parameters were surprisingly the same regarding financial and medical problems. Both were optimized at window length = 10 and prediction range = 5. In this regard, the optimal selection is suggested to explore in future work whether different circumstances have the same model parameters. As for the limitations, the range of window length settings is relatively large. Future research can be carried out within 10. Besides, the research objects are limited to Bitcoin and Ethereum, and more cryptocurrencies can be introduced for experimental modelling. Moreover, in the case of COVID-19, the data selection is only limited to mainland China which might not be representative, since the number of diagnosed cases is small. However, by comparing the model performance between financial and medical issues, the LSTM model parameter settings are suggestive in future work in a wide range of research pathways. Future work on the LSTM model application should focus on multiple combinations of the window length and prediction range parameters. It is advised to take the research results of window length = 10 and prediction range = 5 as a parameter setting cut-off to conduct comprehensive work in multiple research areas.

## 8. Additional Points

*Highlights.* The study uses different combinations of window sliding and prediction range settings to improve LSTM model. The study combines the Random Walk model and LSTM model based on economic theory to conduct experiment. By the insight of parameters settings in the financial case, the study applied the same parameters in the medical issues and verified the performance in both circumstances. The study proposes a view about the balance of model's accuracy and practicality based on the comparison of financial issues and medical issues. *Paraphrase.* Some of the ideas come from the author's master's dissertation in University of Southampton, which might be with some similarity in Turnitin. The supervisor agreed to use this idea as it is not a formal publishing.

## Data Availability

The data used to support the findings of this study are included within the article.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] A. ElBahrawy, L. Alessandretti, A. Kandler, R. Pastor-Satorras, and A. Baronchelli, *Evolutionary dynamics of the cryptocurrency market*, *Royal Society open science*, vol. 4, Article ID 170623, 2017.
- [2] L. Alessandretti, A. ElBahrawy, L. M. Aiello, and A. Baronchelli, *Machine learning the cryptocurrency market*, SSRN, London, UK, 2018.
- [3] I. Kaastra and M. Boyd, "Designing a neural network for forecasting financial and economic time series," *Neurocomputing*, vol. 10, pp. 215–236, 1996.
- [4] H. White, *Economic prediction using neural networks: The case of IBM daily stock returns*, vol. 2, pp. 451–458, ICNN, 1988.
- [5] B. G. Malkiel, "Models of stock market predictability," *Journal of Financial Research*, vol. 27, pp. 449–459, 2004.
- [6] A. W. Lo, *Efficient Markets Hypothesis*, Investopedia, New York, NY, USA, 2007.
- [7] J. L. Elman, "Finding structure in time," *Cognitive science*, vol. 14, pp. 179–211, 1990.
- [8] S. McNally, J. Roche, and S. Caton, "Predicting the price of bitcoin using machine learning," in *Proceedings of the 2018 26th IEEE International Conference on Parallel, Distributed and Network-Based Processing (PDP)*, pp. 339–343, IEEE, Cambridge, UK, March 2018.
- [9] S. J. Brown, W. N. Goetzmann, and A. Kumar, "The dow theory: william peter Hamilton's track record reconsidered," *The Journal of Finance*, vol. 53, no. 4, pp. 1311–1333, 1998.
- [10] V. S. Ediger and S. Akar, "Arima forecasting of primary energy demand by fuel in turkey," *Energy policy*, vol. 35, pp. 1701–1708, 2007.
- [11] G. H. Chen, S. Nikolov, and D. Shah, "A latent source model for nonparametric time series classification," 2013, <https://arxiv.org/abs/1302.3639>.
- [12] T. Xia, Y. Song, Y. Zheng, E. Pan, and L. Xi, "An ensemble framework based on convolutional bi-directional LSTM with multiple time windows for remaining useful life

- estimation,” *Computers in Industry*, vol. 115, Article ID 103182, 2020.
- [13] Y. Liu, Y. Wang, X. Yang, and L. Zhang, “Short-term travel time prediction by deep learning: a comparison of different lstm-dnn models,” in *Proceedings of the 2017 IEEE 20th international conference on intelligent transportation systems (ITSC)*, pp. 1–8, IEEE, Yokohama, Japan, October 2017.
- [14] R. J. Barro and X. Sala-iMartin, “Convergence,” *Journal-ofpoliticalEconomy*, vol. 100, pp. 223–251, 1992.
- [15] A. M. Antonopoulos, *MasteringBitcoin:unlockingdigitalcryptocurrencies*, O’ReillyMedia, Inc, Sebastopol, California, 2014.
- [16] M. Haferkorn and J. M. Q. Diaz, “Seasonality and interconnectivity within cryptocurrencies-an analysis on the basis of bitcoin, litecoin and namecoin,” in *Proceedings of the International Workshop on Enterprise Applications and Services in the Finance Industry*, pp. 106–120, Springer, Sydney, Australia, December 2014.
- [17] W. Guo, N. Xiong, H.-C. Chao, S. Hussain, and G. Chen, “Design and analysis of self Adapted task scheduling strategies in wireless sensor networks,” *Sensors*, vol. 11, no. 7, pp. 6533–6554, 2011.
- [18] X. Wang, Q. Li, N. Xiong, and Y. Pan, “Ant colony optimization-based location-aware routing for wireless sensor networks,” in *Proceedings of the International Conference on Wireless Algorithms, Systems, and Applications*, pp. 109–120, Springer, Dallas, TX, USA, October 2008.
- [19] R. Wan and N. Xiong, “Anenergy-efficientsleepschedulingmechanismwithsimilaritymeasureforwirelessensornetworks,” *Human-centric Computing and Information Sciences*, vol. 8, pp. 1–22, 2018.
- [20] Y. Lu, S. Wu, Z. Fang, N. Xiong, S. Yoon, and D. S. Park, “Exploring finger vein based personal authentication for secure iot,” *Future Generation Computer Systems*, vol. 77, pp. 149–160, 2017.
- [21] Y. Jiang, G. Tong, H. Yin, and N. Xiong, “A pedestrian detection method based on genetic algorithm for optimize xgboost training parameters,” *IEEE Access*, vol. 7, Article ID 118310, 2019.
- [22] L. A. Crosby, K. R. Evans, and D. Cowles, “Relationship quality in services selling: an interpersonal influence perspective,” *Journal of Marketing*, vol. 54, no. 3, pp. 68–81, 1990.
- [23] A. Graves, “Sequencetransductionwithrecurrentneuralnetworks,” 2012, <https://arxiv.org/abs/1211.3711>.
- [24] A. Urquhart, “Theinefficiencyofbitcoin,” *EconomicsLetters*, vol. 148, pp. 80–82, 2016.
- [25] K. Pearson, “Theproblemoftherandomwalk,” *Nature*, vol. 72, p. 294, 1905.
- [26] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, “Normalizing flows for probabilistic modeling and inference,” *Journal of Machine Learning Research*, vol. 22, pp. 1–64, 2021.
- [27] T. Chai and R. R. Draxler, “Root mean square error (rmse) or mean absolute error (mae)?–arguments against avoiding rmse in the literature,” *Geoscientific Model Development*, vol. 7, no. 3, pp. 1247–1250, 2014.
- [28] P. Zhang, “Ontheconvergencerateofmodelselectioncriteria,” *CommunicationsinStatistics-TheoryandMethods*, vol. 22, pp. 2765–2775, 1993.
- [29] X. Yuan, C. Chen, M. Jiang, and Y. Yuan, “Prediction interval of wind power using parameter optimized beta distribution based lstm model,” *Applied Soft Computing*, vol. 82, Article ID 105550, 2019.
- [30] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, “Lstm-based encoder-decoder for multi-sensor anomaly detection,” 2016, <https://arxiv.org/pdf/1607.00148>.
- [31] Z. Wang, X. Su, and Z. Ding, “Long-term traffic prediction based on lstm encoder-decoder architecture,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 10, pp. 6561–6571, 2021.

## Research Article

# Novel Crow Swarm Optimization Algorithm and Selection Approach for Optimal Deep Learning COVID-19 Diagnostic Model

**Mazin Abed Mohammed** <sup>1</sup>, **Belal Al-Khateeb**,<sup>1</sup> **Mohammed Yousif**,<sup>2</sup>  
**Salama A. Mostafa** <sup>3</sup>, **Seifedine Kadry** <sup>4</sup>, **Karrar Hameed Abdulkareem** <sup>5,6</sup>  
and **Begonya Garcia-Zapirain**<sup>7</sup>

<sup>1</sup>College of Computer Science and Information Technology, University of Anbar, Ramadi 31001, Anbar, Iraq

<sup>2</sup>Directorate of Regions and Governorates Affairs, Ministry of Youth & Sport, Ramadi 31065, Anbar, Iraq

<sup>3</sup>Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Johor 86400, Malaysia

<sup>4</sup>Department of Applied Data Science, Noroff University College, Kristiansand 4608, Norway

<sup>5</sup>College of Agriculture, Al-Muthanna University, Samawah 66001, Iraq

<sup>6</sup>College of Engineering, University of Warith Al-Anbiyaa, Karbala, Iraq

<sup>7</sup>eVIDA Laboratory, University of Deusto, Avda/Universidades 24, Bilbao 48007, Spain

Correspondence should be addressed to Mazin Abed Mohammed; [mazinalshujeary@uoanbar.edu.iq](mailto:mazinalshujeary@uoanbar.edu.iq)

Received 1 January 2022; Revised 16 March 2022; Accepted 19 July 2022; Published 13 August 2022

Academic Editor: Mohammed A. A. Al qaness

Copyright © 2022 Mazin Abed Mohammed et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Due to the COVID-19 pandemic, computerized COVID-19 diagnosis studies are proliferating. The diversity of COVID-19 models raises the questions of which COVID-19 diagnostic model should be selected and which decision-makers of healthcare organizations should consider performance criteria. Because of this, a selection scheme is necessary to address all the above issues. This study proposes an integrated method for selecting the optimal deep learning model based on a novel crow swarm optimization algorithm for COVID-19 diagnosis. The crow swarm optimization is employed to find an optimal set of coefficients using a designed fitness function for evaluating the performance of the deep learning models. The crow swarm optimization is modified to obtain a good selected coefficient distribution by considering the best average fitness. We have utilized two datasets: the first dataset includes 746 computed tomography images, 349 of them are of confirmed COVID-19 cases and the other 397 are of healthy individuals, and the second dataset are composed of unimproved computed tomography images of the lung for 632 positive cases of COVID-19 with 15 trained and pretrained deep learning models with nine evaluation metrics are used to evaluate the developed methodology. Among the pretrained CNN and deep models using the first dataset, ResNet50 has an accuracy of 91.46% and a F1-score of 90.49%. For the first dataset, the ResNet50 algorithm is the optimal deep learning model selected as the ideal identification approach for COVID-19 with the closeness overall fitness value of 5715.988 for COVID-19 computed tomography lung images case considered differential advancement. In contrast, the VGG16 algorithm is the optimal deep learning model is selected as the ideal identification approach for COVID-19 with the closeness overall fitness value of 5758.791 for the second dataset. Overall, InceptionV3 had the lowest performance for both datasets. The proposed evaluation methodology is a helpful tool to assist healthcare managers in selecting and evaluating the optimal COVID-19 diagnosis models based on deep learning.

## 1. Introduction

In December 2019, a new coronavirus called (COVID-19) was appeared in China, particularly in Wuhan. The COVID-19 spread worldwide caused disastrous effects and lead to death [1]. COVID-19 pandemic gets great attention from

global and health institutions as it has no cure yet [2]. COVID-19 consists of RNA-type with positive-oriented single-stranded, making finding the treatment challenging because of the mutating characteristics [3]. Scientists and researchers have created a hard effort worldwide to discover an effective treatment for COVID-19. According to the

global statistical data, China, the USA, Brazil, Italy, Spain, Iran, the UK, and many other countries have lost thousands of persons due to the COVID-19 [4]. The coronavirus family has different types, and they can infect animals. This issue includes COVID-19, which can be seen in bats, poultry, rodent, cat, dog, pigs, and humans. The common COVID-19 symptoms are fever, dry cough, headache, sore throat, runny nose, and muscle pains. Infected people with COVID-19 who have weak immune systems are the potential to die. COVID-19 can infect healthy people through physical contact such as breath, mucous, and hand contact with infected people [5].

Different diagnosis biomarkers have been used in the detection of COVID-19, such as blood test samples, X-ray, breathing recordings, ultrasound images, nuclear medicine imaging, and CT images [1]. For the regions that were early attacked by the COVID-19 pandemic, RT-PCR can be an inappropriate examination. As reported in [6], some lab COVID-19 examinations are deficient in sensitivity with 71%, which is based on many factors, including quality control and sample preparation. Other radiology examinations, including thoracic CT and lung X-ray, positively help the medical practitioners in COVID-19 diagnosis [7]. Many cases in China diagnosed with COVID-19 have shown some abnormalities in the CT scans [8]. People who are suspected of COVID-19 infection with no apparent symptoms ask to quarantine to make more COVID-19 examinations. In the context of examination sensitivity, people who are suspected of COVID-19 infection must make a nucleic acid test several times to confirm the COVID-19. The results of imaging examinations are essential to combat the COVID-19 slowdown it is spread. Radiological imaging is the most rapid, accurate examination for COVID-19 diagnosis. By using computed tomography (CT) imaging, to detect any abnormalities, most of the cases share the same characteristics, including a rounded morphology and a peripheral distribution. Also, pulmonary consolidation appears in the advanced stage, and ground-glass opacities appear in the early stage [7]. CT imaging examination can assist radiologists and doctors in diagnosing COVID-19 early. However, sometimes radiologists and doctors find it difficult to diagnose COVID-19 based on CT images as the viral cases of pneumonia look like other inflammatory diseases in the lung. Concerning imaging-based COVID-19 examination, particularly CT imaging, there are three diagnosis workflow stages: pre-scan preparation stage, acquisition of image stage, and final diagnosis of disease stage. In the first stage, the patient is prepared and instructed for bed examination based on a certain protocol. The CT images are scanned in the image acquisition stage, and the patient is requested to hold their breath during the scan process, which covers the area of the upper lung through the lung base. The radiologists set the proper measurements of the area that would be covered in the scan based on the body shape of the patient. CT images are generated from the obtained raw data. Then they are processed via picture archiving and communication system (PACS) for diagnosis. Artificial intelligence (AI) can positively be incorporated with medical imaging to help in the COVID-19 diagnosis and combat the

disease [9]. In contrast to the classical imaging tool procedures, which mainly depend on technicians and radiologists, imaging tools incorporated with AI are less human dependent and more accurate, safer, and efficient. The modern AI-driven system for COVID-19 diagnosis has a dedicated imaging platform, segmentation of the infected areas, and various diagnosis and evaluation tools. Furthermore, AI technology is embedded successfully in various commercial systems to fight COVID-19 and use this advanced technology. In February 2020, the first virtual seminar on COVID-19 was organized by the Medical Imaging Computing Seminar (MICS), an alliance of medical imaging scholars and start-up companies in China. Thousands of interested people attended the seminar. Those examples indicate how strong the public interest regarding the utilization of the AI in the imaging field for medical purposes to fight COVID-19.

Optimization methods and AI models have a great capability to combat COVID-19 by obtaining more accurate and reliable diagnoses in a short and optimized time efficiently. In recent times, many AI-based computer-assisted systems have been used in various medical institutions and hospitals to diagnose COVID-19 automatically rather than the traditional manual method of data analysis. Although many AI methods are utilized in automated COVID-19 systems, it is a big challenge for medical institutions and hospitals to select the best method that meets their needs and produces reliable and accurate results [10]. However, there is no AI-based model better than others [11]. The challenge becomes more prominent when the hospitals and healthcare managers need to evaluate the AI-based model with various metrics. Furthermore, many deep learning detection models are designed for COVID-19 diagnosis. Healthcare managers face difficulty selecting the appropriate method, evaluating it with different metrics, and validating the medical solutions. A wrong solution has devastating effects that might lead to losing a patient's life, financial crisis, and legal accountability. For instance, if the AI-based model obtains the wrong result and shows a positive COVID-19 for the noninfected healthy person, the person will receive unnecessary treatment with side effects. In contrast, if the AI-based model obtains the wrong result and shows a negative COVID-19 for the infected person, the patient will not receive the correct cure, and their health condition might be worsening. In addition, the patient infects the other noninfected people. Both examples have severe consequences for the hospitals and healthcare institutions concerning credibility and reputation. Therefore, selecting a diagnostic AI model that produces free errors, reliable, and non-costly solutions is therefore essential. Moreover, making an evaluation is not a trivial task, especially when various measurements are involved. Group reliability and time complexity are two popular criteria that must be considered to evaluate deep learning models of COVID-19 diagnosis. Concerning group reliability, many measures can belong to group reliability, including *F1*-score, precision, average accuracy, recall, error rate, true negative (TN), false negative (FN), true positive (TP), and false positive (FP) [12]. According to [13], the accuracy criterion has been used to evaluate various deep learning models, such

as studies in [14]. The evaluation of the deep learning model for COVID-19 diagnosis is not only restricted to accuracy criteria. There are multiobjective/criteria that must be considered in the evaluation when selecting the optimal model.

This study proposes an integrated evaluation methodology for different COVID-19 diagnosis models based on deep learning. The method motivates the authors to develop integrated deep learning classifiers under one framework and involves the most common performance evaluation criteria of COVID-19 diagnosis models based on deep learning. The proposed evaluation methodology is a helpful tool to assist healthcare managers in selecting and evaluating the optimal COVID-19 diagnosis models based on deep learning. The main contributions of this study can be summarized as follows:

- (i) An integrated method is proposed for selecting the optimal deep learning model based on the novel crow swarm optimization (CSO) algorithm for COVID-19 diagnosis.
- (ii) The crow swarm optimization (CSO) is employed to find an optimal set of coefficients using a designed fitness function for evaluating the performance of the deep learning model. The CSO is modified to obtain a good distribution of selected coefficients by considering the best average fitness.
- (iii) In order to show its performance, CSO is benchmarked with some well-known swarm optimization algorithms, those are Grey Wolf Optimization (GWO), Harris Hawks Optimization (HHO), Salp Swarm Algorithm (SSA), and Whale Optimization Algorithm (WOA).
- (iv) We have utilized two datasets: the first one includes 746 CT images, 349 of which are of confirmed COVID-19 cases and the other 397 are of healthy individuals; and the second dataset composed of unimproved CT images of the lung for 632 positive cases of COVID-19 with 15 trained and pretrained deep learning models with nine evaluation metrics used to evaluate the proposed methodology.
- (v) Deep convolutional CNN feature representation is applied to extract highly representative features via successful descriptors of deep CNN. Our proposed method can distinguish between the infected region of COVID-19 and the noninfected region in the lung CT scan and X-ray images, which improves the accuracy compared to other existing methods. To the best of the authors' knowledge, this work is the first inclusive study, including 15 deep learning classifiers.
- (vi) The devolved method minimizes the classification time significantly while yielding higher accuracy. This is a great benefit when developing an automatic real-time medical system.

The rest of this article is organized as follows: Section 2 presents the related works of the COVID-19 diagnostic systems highlighting the initial associated studies. The

developed selection methodology for COVID-19 diagnostic models is proposed in Section 3. The experimental result of the proposed COVID-19 diagnostic selection methodology based on the CSO algorithm is provided in Section 4, with the study limitation and future work. Finally, conclusions have been provided in Section 5.

## 2. Related Works

COVID-19 is caused by severe acute respiratory syndrome corona virus 2 (SARS-CoV-2). It first appeared in Wuhan city in China in late 2019 [15]. As a result of the COVID-19 pandemic, scientists and researchers in the medical and healthcare community are dedicating efforts to finding a solution to fight COVID-19 and control the spreading rate [16]. Several COVID-19 related studies are proposed to assess pneumonia caused by COVID-19 and the degree of COVID-19 infection to make the appropriate decision regarding the treatment plan and select the appropriate medication and required doses. Hospitals and medical centers widely used noninvasive image methods such as X-ray and lung CT scans to detect COVID-19 pneumonia severity. The result of CT images is more precise compared to X-rays. Therefore, this work is limited to the CT scan imaging (CTSI) examination only. Moreover, the CTSI provides more accurate results than reverse transcription-polymerase chain reaction (RT-PCR). There are massive studies presented recently that proposed detection and prediction methods for COVID-19 diagnosis [17]. In [4], a deep review of the prediction and detection methods for COVID-19 is presented. The review also includes laboratory-level detection methods such as RT-PCR and CTSI physical assessment reported by skilled radiologists. The automatic detection method produces a rapid and accurate solution with no need for a hard-human labor effort.

In [18], a deep learning-based method, namely VIDX-Net, is developed for COVID-19 diagnosis using chest X-ray images. Comparative study of various deep learning classifiers including DenseNet and other models is discussed in detail for esNetV2, MobileNetV2, and InceptionV3 with proposed images dataset provided for public use. It consists of 50 X-ray images, half of them belong to the health cases while the other half belong to the COVID-19-infected cases [19]. The result in [20] shows DenseNet201 and VGG19 classifiers perform better than other classifiers with an accuracy of up to 90.00%. A COVID-19 diagnostic model based on machine learning is proposed in [21]. The proposed model used 150 CT images divided into different groups, including 16, 32, 48, and 64. The study also uses various handcrafted features, including discrete wavelet transform (DWT), grey level co-occurrence matrix (GLCM), grey level size zone matrix (GLSZM), local directional pattern (LDP), and grey level run length matrix (GLRLM). A support vector machine (SVM) classifier is employed in the study of [21]. SVM was inserted the extracted features based on different cross-validations (2-fold, 5-fold, and 10-fold). The extractor of GLSZM features achieved a higher accuracy, which equals 98.77% of 10-fold cross-validation. A deep learning-based COVID-Net method using lung X-ray images for the



detection of COVID-19 is presented in [22]. The proposed method's structure combined one-one convolutions, the residual modules, and depthwise convolution to allow a deeper architecture and overcome the gradient vanishing problem. The proposed method used a combination of COVID-19 lung X-ray dataset obtained from [23] with different classification groups, including normal class, viral infection (non-COVID-19), COVID-19, and bacterial infection. The accuracy of the method reached up to 83.5%.

An automatic COVID-19 diagnostic approach based on deep learning and transfer learning strategy is proposed in [24]. The proposed approach's structure combines a convolutional neural network (CNN) and a modified AlexNet [25] with the feasibility of transfer learning. The CNN architecture comprises one convolutional layer, batch normalization with 16 filters, two fully connected layers, and a rectified linear unit (ReLU). The proposed combination approach can obtain accuracy up to 94.00%. An investigation of interpretability of deep learning-based models and uncertainty of COVID-19 diagnostic detection using X-ray images is conducted in [26]. Bayesian convolutional neural network (BCNN) and drop weight mechanism are used to estimate deep learning uncertainty. This combination can increase the clinical practice trust by achieving consistent results. The proposed method to assess the correlation between uncertainty and accuracy uses 70 chest X-ray images of COVID-19 cases which are obtained from public datasets of [23]. The dataset is prepared, and the size of all images is adjusted to 512 pixels. A real-time data augmentation strategy and transfer learning strategy are used to deal with the limitation of the dataset size. The proposed approach is achieved accuracy up to 94.00% when applied VGG16 deep learning method. A transfer learning strategy of a 10-fold cross is combined with a VGG16 architecture in [27]. The proposed model trained using the dataset obtained from [19]. The size of all images is adjusted to 224 pixels. The real-time data augmentation strategy is used to overcome the limitation of the dataset size. The proposed method obtained 96.1% and 99.70% in accuracy and area under the curve (AUC), respectively. A new architecture of fine-tuned and pretrained ResNet50 for COVID-19 is proposed in [28]. The proposed model involved various data augmentation strategies, including random rotation and vertical flip, to enhance the training model generalization. The proposed model yielded accuracy up to 96.23% on a multiple class, including COVID-19 infection, normal, viral infection, and bacterial infection dataset.

For controlling the spread, an initial diagnosis of alleged COVID-19 cases and screenings must be carried out daily as shown in related COVID-19 diagnosis reviews or literature. For a rapid and accurate diagnosis of COVID-19, extraction of radiological features using AI and ML has proven the principle's efficacy as observed in the outcomes. In carrying out clinical diagnoses, the use of X-ray and computed tomography (CT) images essentially provide useful information. This necessitates, therefore, that doctors have an automatic CT image diagnosis system developed to help them in COVID-19 diagnosis. According to [29], a two-stage data enhancement method must be used in classifying the

images of coronavirus and five other situations. Due to unbalanced and deficient image numbers in the dataset, the initial phase applied the use of a shallow image augmentation method. Feature extraction using handcrafted approaches come in more useful and convenient in analyzing these images because of the insufficiency of the newly created dataset in deep architecture training. Furthermore, in the study, the next data enhancement phase utilizes an algorithm called the synthetic minority over-sampling method. Conclusively, a stacked autoencoder and principal component evaluation technique is applied to resize the feature vector  $y$  by removing interlinked or associated features present in the feature vector. As in the obtained results, it is observed that COVID-19 diagnosis can be performed effectively and quickly due to the proposed model's performance leveraging capability.

COVID-19 and other atypical and viral (non-COVID-19) respiratory diseases appear indistinguishable in comparison. A clinical computer-aided diagnosis (CAD) system in the study by [30] applies automatic discrimination of COVID-19 from non-COVID-19 pneumonia patients using CT features. A total of 612 recruitment containing 306 COVID-19 and 306 non-COVID-19 cases were made. From the CT images, extraction of 20 radiological features was performed, and these features were used for the pattern, location, and lesions' distribution evaluations of patients in groups. To evaluate the CAD system with best performance and classification of COVID-19 and non-COVID-19 cases, support vector machines, naïve Bayes, decision tree, k-nearest neighbor, and ensemble are five classifiers trained using all the significant CT features. These significant COVID-19 groups are air bronchogram, cavity, consolidation, crazy-paving, and ground-glass opacity (GGO), involvement distribution pattern and location, lesion numbers, lymphadenopathy, nodule, pleural effusion, and thickening, reticular, and thickening of the bronchial wall. On implementation using an ensemble COVIDiag classifier, an accuracy of 91.94%, a sensitivity of 0.965, and specificity of 93.54% were observed in the proposed CAD system. A COVIDiag model, as suggested by this study using CT radiological routine features, provided results that are encouraging in COVID-19 diagnosis. The study claims that radiologists can consider using this tool as support in making better and accurate COVID-19 diagnoses in this present pandemic. [31] came up with a diagnosis framework called CovidCTNet in an attempt to better CT imaging detection accuracy. CovidCTNet comprises a set of deep learning algorithms that accurately distinguish COVID-19 from any community-acquired pneumonia (CAP) or other respiratory ailments. CovidCTNet obtained an accuracy of 95% in CT imaging detection as to 70% obtained from radiologists. Independent of the CT imaging device, CovidCTNet is embedded with the ability to work with heterogeneous and small sample size data. [31] made available in open source the model metrics and all algorithms in detail to more trustworthy the detecting capacity of COVID-19 globally and support radiologists and doctors during the screening procedures. While CovidCTNet's sharing helps to preserves data ownership and user confidentiality, it further

facilitates rapid improvement and service optimization by developers. Another study by [32] concentrated on applying different deep learning methods to distinguish between COVID-19 and non-COVID-19 CT scan images. As a result, a CTnet-10 model with an accuracy of 82.1% was self-developed and designed. Furthermore, other models including DenseNet169, InceptionV3, ResNet-50, and VGG19 were tested. VGG19 came up with a superior accuracy of 94.52% among the trialed deep learning techniques from the results. For rapid and effective screening for COVID-19 diagnosis, automated COVID-19 diagnosis from CT scan images can prove a useful method to doctors.

In the work of [33], 1065 CT pictures of previously diagnosed patients with common viral pneumonia and pathogen-confirmed COVID-19 cases were obtained. The inception transfer learning model was modified to develop the algorithm and further carried out the internal and external validation. An accuracy of 89.5%, specificity of 88.0%, and sensitivity of 87.0% were obtained for the internal validation. In the external validation, the accuracy, specificity, and sensitivity showed 79.3%, 83.0%, and 67.0%, respectively. More so, the algorithm with 85.2% accuracy detected 46 out of 54 COVID-19 images as COVID-19 positive, with the first two nucleic acid test outcomes were negative. Similarly, the work of [34] obtained CT images of 262 persons for COVID-19, 100 persons for bacterial pneumonia, 219 persons for common viral pneumonia, and 78 persons for healthy control. They combined the newly developed ResNet50 backbone and SE blocks for image analysis to come up with a model that can effectively detect and obtain the indefinite or abstruse differences in CT images. This model produced accuracy, AUC, recall, precision, and  $F1$ -score of 94%, 0.96, 0.94, 0.95, and 0.94, respectively, which shows superior performance compared to the generally utilized basic models.

Based on this review for the recent COVID-19 diagnosis deep learning or machine learning methods using CT or X-ray image classification, several open research issues need further studies. The first issue is that there is no standard and certified CT or X-ray image dataset with the quality and quantity of the images to produce reliable results. The second issue is there is no deep learning or machine learning model that can yield the best results. Each of the proposed models might produce acceptable to high results in certain circumstances. The third issue is determining the main constraint of the existing models that affect the performance and the diagnosis quality. Most AI-based COVID-19 are not available for public use to the best of the authors' knowledge, which does not allow the researchers to use them in the research. However, some other researchers attempt to make COVID-19 radiography imaging and COVID-19 diagnosis model based on AI and deep learning that are accessible for the research community. The COVID-19 chest CT and X-ray dataset are publicly available now. Evaluation of the deep learning-based models with the group reliability and the time complexity of time is a necessity to assess their quality. ML-based models that used radiography can produce more accurate and reliable results [35].

[36] propose a classification method of COVID-19 based on CT images. The method integrated CNN with transfer learning and sparrow search algorithm (SpaSA) for hyperparameter optimization. The transfer learning is used to initialize the CNN training cycle, and the SpaSA is used to select the best trained model. The method has achieved the best accuracy of 99.74%. Also, the study made by [37] focuses on identifying the state-of-the-art computational algorithms for diagnosing COVID-19. The study claims that CT images can provide the means for the early detection of COVID-19. Moreover, the computational algorithms' ability to detect COVID-19 is highly affected by detecting certain visual features in the CT images, such as the ground-glass opacity (GGO) feature. The study concludes with the need for the initial training of the deep learning algorithms to overcome the unavailability of sufficient training samples. Another work by [38] propose a CNN model with depthwise separable dense and convolution block attention module. The convolutional block attention module is used to extract high-quality features that help to overcome the overfitting of the training model. The depthwise separable dense is used to reduce the dimensionality of the features and support lightweight prediction models. The model has been tested on a small sample of X-ray and CT images and is able to achieve an accuracy of 98.62% and 99.18, respectively. It is able to reduce training parameters and outperforms four similar models. The work of [39] attempts to evaluate the performance of four machine learning models, namely decision tree (DT), partial least squares discriminant analysis (PLS-DA), artificial neural network (ANN), and K-nearest neighbor algorithm (KNN), in COVID-19 diagnosis and severity. The used data for testing are laboratory tests results of patients (557 positives and 5,086 negatives COVID-19). The accuracy of the four tested models has exceeded 84%, and the ANN model achieves the best performance of 96% accuracy on average.

In the literature, numerous studies are presented that deal with radiography imaging for COVID-19 diagnosis. But no such study deals with evaluating the diagnostic system based on deep learning models to assist the healthcare managers in selecting the optimal COVID-19 diagnostic system. This work attempts to bridge the gap between the selections of COVID-19 diagnostic ML-based model and radiography imaging. This study proposed automatic COVID-19 detection deep learning-based models using chest CT images. This motivates the authors to employ 15 deep learning models, including MobileNets V2, VGG19, DarkNet, ResNet50, Xception, GoogleNet, ResNet34, SAE, CNNs, InceptionResNetV2, NASNet-Large, InceptionV3, LSTM, and DNN, to find the optimal accuracy using 746 chest CT images dataset. Also, in this study, we compute 9 evaluation measurements, including classification accuracy rate (CAR), predictive positive value (PPV),  $F1$ -score, false positive rate (FPR), mean squared error (MSE), precision, AUC (area under the curve), negative predictive values (NPV), recall, and ROC (receiver operating characteristics) curve. The crow swarm optimization (CSO) is employed in this study to find an optimal set of coefficients based on a designed fitness function for the evaluation the deep

learning performance. CSO acts as a distribution strategy to ensure all the selected coefficients are distributed fairly by considering only the best available average fitness.

### 3. Methodology

Rapid and correct diagnosis of COVID-19 possible cases performs a vital role in quarantine and treatment systems. Exact quick, exceptionally early diagnosis results of COVID-19 suspected cases plays a significant part in inconvenient isolation and treatment, which is also of extraordinary significance for patients' guesses, the control of this scourge, and the open well-being security. But right now, a huge number of suspected patients must experience chest CT checking. This process has caused a huge burden to proficient therapeutic staff. Their extreme deficiency is also a major challenge within the current circumstance; additionally, radiologists' visual weariness would increase the potential dangers of failure in diagnosing all the cases. Developing a computerized detection system based on CT lung scan images is valuable to counter the outbreak of COVID-19 or SARS-CoV-2. The previous studies like [40] show that with a considerable small sample of scanned CT lung scan images between 500 and 800, different types of DL algorithms are able to detect COVID-19 cases with high accuracy [41]. Subsequently, this research proposes an automatized deep learning-based COVID-19 detection system. The study aims to investigate the most optimal DL model for a more effective COVID-19 detection system in comparison to the latest DL computer-aided diagnosis methods. The proposed selection method for the best DL models is shown in Figure 1. Also, pseudocode for the selection approach for optimal deep learning COVID-19 diagnostic model using CSO approach is presented in Algorithm 1. The methodology section is divided into three main sections as follows:

#### 3.1. Development Stage

**3.1.1. CT Lung Scan COVID-19 Dataset.** In this study, we have utilized two datasets to validate our methods. The first dataset includes 746 CT images, 349 of them are confirmed COVID-19 cases and the other 397 are of healthy individuals [19]. The CT images and other references and resources are mainly taken from free and open access websites such as medRxiv and bioRxiv. The primary COVID-19 dataset collection of our work covered the period from January 19 to May 25 2020. The COVID-19 cases of the dataset comprise a full clinical depiction of the patients' conditions. A case of chest CT images for patients having COVID-19 is presented in Figure 2.

The second dataset is the NIfTI retroactive dataset composed of the lung's unimproved CT images for 632 positive cases of COVID-19. These images are obtained from a medical care center for people who show a reverse transcription-polymerase chain reaction (RT-PCR) to confirm the COVID-19 infections at the rapid spread of the corona pandemic. Patient severing with some symptoms of COVID-19 are caused by direct contact with an infected people or travel to countries affected by Corona. Initial CT

examination of the suspected patients confirms the positive COVID-19 case based on RT-PCR. A soft tissue rebuilding algorithm is used to examine the CT images with no need for vascular contraindication. At a subsequent time, all DICOM images are transformed into the format of NIfTI [42].

**3.1.2. Deep Learning Models.** Recently, one of the tremendously expanding machine learning algorithms in medical imaging research is DL [43]. It has achieved significant diagnostic outcomes in different disease detection types, including cancers and brain, heart, and lung diseases [44]. The different image-based datasets such as ImageNet introduces millions of images as training and testing dataset [45]. For instance, in 2020 [46], DL models show dermatologist-level execution on classifying skin lesions. In 2018, the DL model of [47] produced exceptional outcomes for diagnosing breast cancer from image screening. However, in many cases, the most profound learning-based algorithms for infection determination require explaining the lesions, particularly for infection location in the CT volumes. There are several breakthroughs of DL neural network models that outperform human-level execution. Subsequently, in this study, 15 DL models are selected to investigate the best suitable model for COVID-19 diagnosis comprehensively. The models are convolution neural network (CNN), DarkNet, deep neural network (DNN), GoogleNet, Inception-ResNetV2, InceptionV3, LSTM, MobileNetV2, NASNet-Large, ResNet34, ResNet50, Stacked autoencoder (SAE), VGG16, VGG19, and Xception. They are briefly described in Table 1 and explained in the following. Also, the parameters of the COVID-19 deep learning models have been listed in Table 2.

- (i) Convolution Neural Network: The convolution neural network (CNN) is a multilayer neural network consisting of a set of fully connected layers and convolution layers. The convolution layers are the standard layers of the CNN. The CNN's basic concept includes perception, weights, and sampling (time or space), which backs to the early 60s [48]. The CNN and the neural network, in general, have a local perception that efficiently detects the local feature of the data or object in an image. The CNN input parameters are usually fewer than the hidden layers, which makes it less data dependent.
- (ii) DarkNet: The DarkNet DL model is designed based on state-of-the-art Darknet-19 architecture. The Darknet-19 type is a classification model that has been made for the YOLO tool to perform real-time object detection. This system has an architecture designed for object detection [49].
- (iii) Deep Neural Network: The deep neural network (DNN) is a type of neural network that conducts intensive computation to its input because of nonlinear transformation in its hidden layers. Unlike the conventional neural network, the hidden layers encompass nonlinear functions for further analysis. The DNN has many hidden nodes

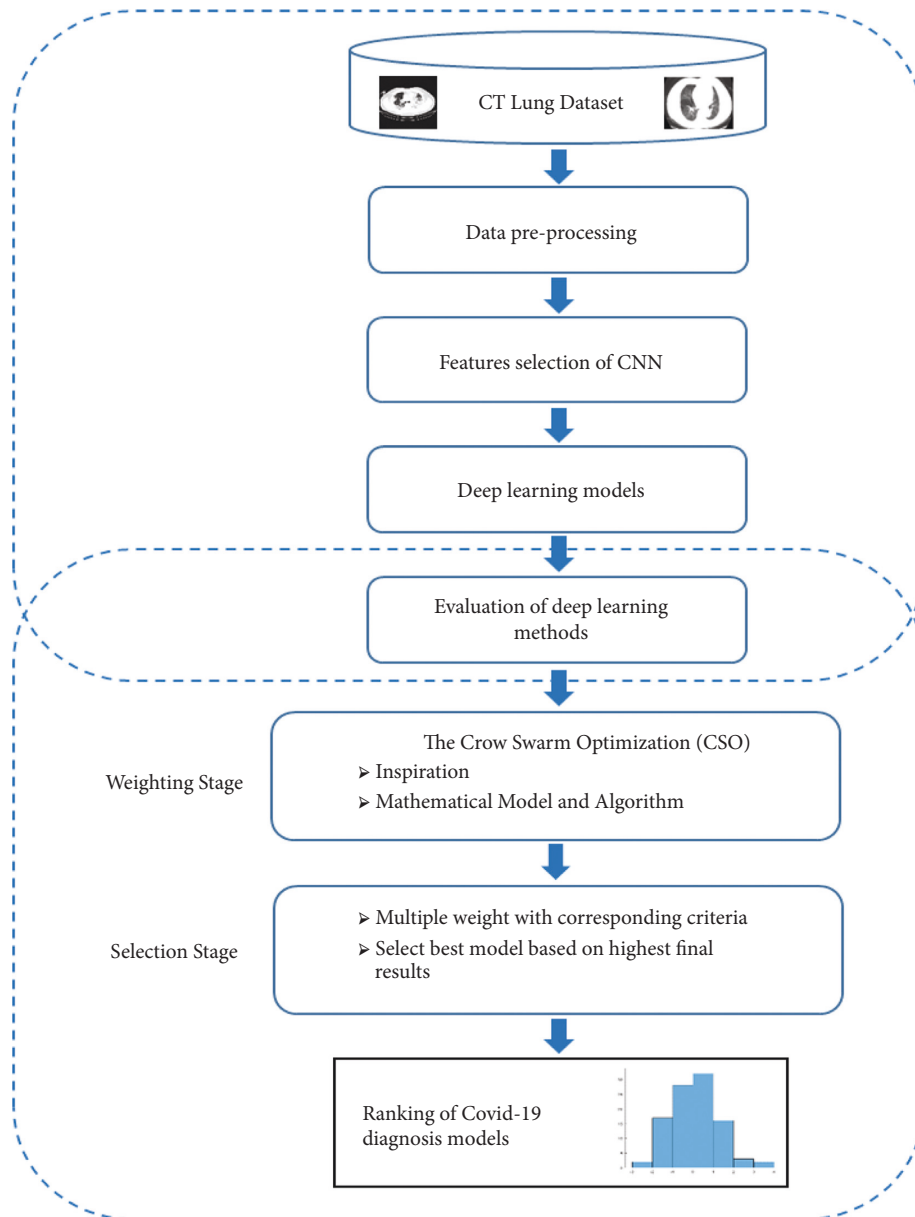


FIGURE 1: The selection approach for optimal deep learning COVID-19 diagnostic model based on novel CSO algorithm.

compared with the conventional neural network. Subsequently, it takes a longer execution time and produces more accurate results [50].

- (iv) GoogleNet: The GoogleNet is a DL model that uses very few input parameters. It is tested on Places365 and ImageNet datasets for object detection and found to be efficient in terms of accuracy and execution time. The GoogleNet model network consists of 22 layers (three convolution layers, nine inception blocks, and the rest are fully connected layers) [51].
- (v) InceptionResNetV2: The InceptionResNetV2 is a CNN model that includes numerous types of pooling and convolution layers. As in many other models, it contains fully connected layers before the output layer. The early InceptionResNet model has been recognized as one of the top CNN models

based on different benchmark datasets such as the ImageNet dataset and JFT Google internal datasets [52]. The InceptionResNetV2 is known for its efficient execution to low-level operations and scalability to fit with a different type of application.

- (vi) Inceptionv3: The Inceptionv3 is the third generation of Google's Inception CNN that initially introduced a module for GoogleNet. It is tested in the ImageNet Recognition Challenge of visual object detection and image analysis. It is very popular in classifying visual objects for computer vision applications [53].
- (vii) LSTM: The LSTM is a type of recurrent neural network (RNN) proposed by Hochreiter and Schmidhuber to deal with sequences of data efficiently. It uses gates to regulate the input data

```

Input: Read CT Lung images with length of H x W
Begin
  Data preprocessing
  Feature selection of CNN
  While ( $I \leq \text{max. epoch}$ )
    Deep learning models
    Evaluation of deep learning methods
    While ( $j < \text{max. iteration}$ )
      CSO mathematical model and algorithm
      Group Division
      Update speed by E.q (7).
      Update position by E.q (8)
      Update angle bt E.q (9)
    End while
    Multiple weight with corresponding criteria
    Selection best model based on highest final results
  Return: Ranking of COVID-19 diagnosis models
Output: Best COVID-19 diagnosis models
End

```

ALGORITHM 1: Pseudocode for the selection approach for optimal deep learning COVID-19 diagnostic model based on novel CSO algorithm.

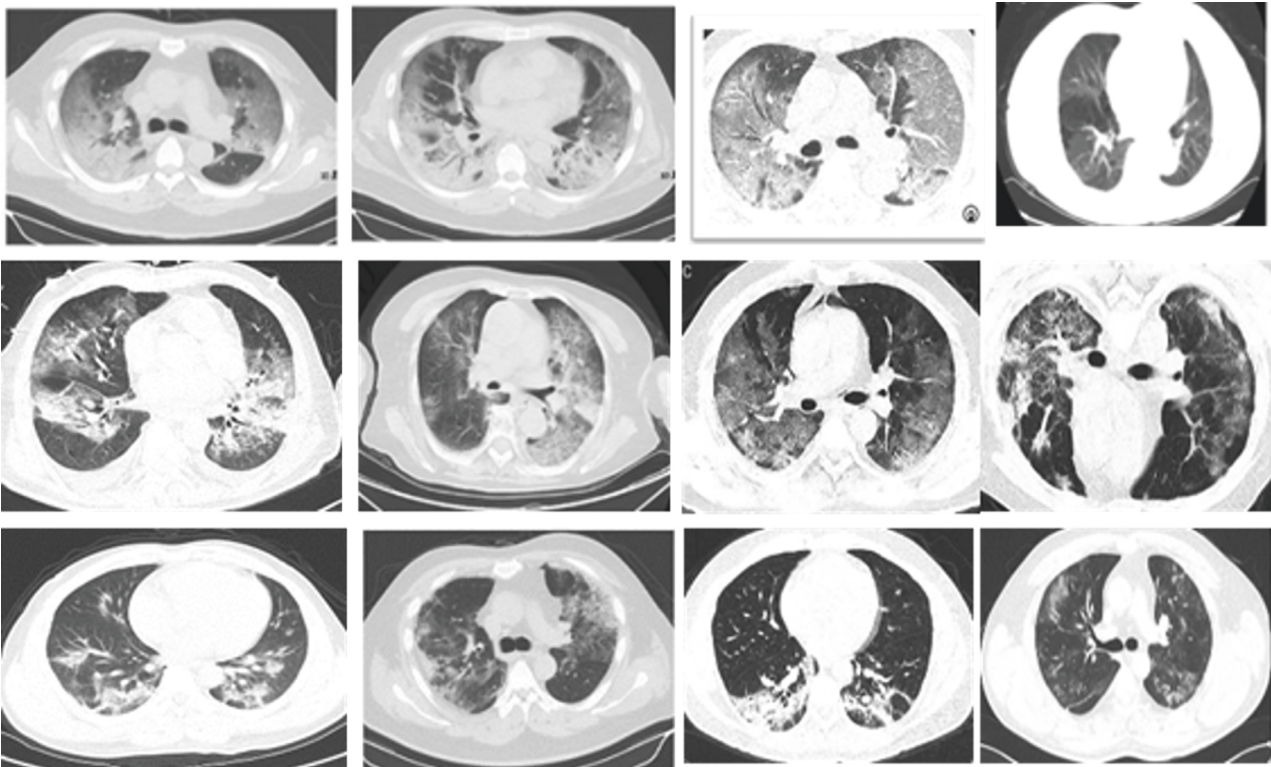


FIGURE 2: COVID-19 CT lung scan cases.

before the learning and remembering process. However, the gates might filter out important data that affect the network's performance [53].

- (viii) MobileNetV2: The MobileNetV2 is the second version of the MobileNetV1 DL neural network included in the TensorFlow-Slim Image Classification, Collaboratory, and some other alternative

libraries. It has lower complexity and model size compared with the MobileNetV1. Google proposes it for mobile phone visual recognition applications to perform object detection, classification, and semantic segmentation [53].

- (ix) NASNet-Large: The NASNet-Large is a CNN model that is trained to classify 1000 objects based

TABLE 1: Deep learning models.

No.	DL model	Description	Remark
1.	CNN	It consists of a set of fully connected layers and convolution layers	Require a few input parameters
2.	DarkNet	Classification model for object detection	Used for real-time object detection
3.	DNN	It has many hidden nodes compared with the conventional neural network	Performs deep nonlinear analysis
4.	GoogleNet	It is an improved DL model for image analysis	Used for object detection with a few input parameters
5.	InceptionResNetV2	It has a fixable architecture of a CNN	Used for different types of applications
6.	Inceptionv3	Third generation of Google's Inception CNN	Used for classifying visual objects for computer vision applications
7.	LSTM	A type of recurrent neural network (RNN)	Used for dealing with sequences of data
8.	MobileNetV2	A lower complexity and model size DL neural network proposed by Google for mobile phone image processing applications	Used for object detection, classification, and semantic segmentation
9.	NASNet-Large	A CNN modeled to deal with a large scale of image datasets.	Used to classify objects
10.	ResNet34	A CNN architecture but with shortcuts and bottleneck block mechanisms between layers to speed up solving problems.	Used for deep real-time analysis
11.	ResNet50	A type of CNN that performs deeper analysis to solve complex problems	The deeper analysis might degrade the accuracy of the network
12.	SAE	A multilayer neural network with a stacked autoencoder	Used for datasets with a small dimension of features.
13.	VGG16	A CNN with multiple $3 \times 3$ kernel-sized filters in the convolutional layers	Used for recognition tasks of a large-scale number of images dataset
14.	VGG19	A CNN with multiple $3 \times 3$ kernel-sized filters in the convolutional layers with additional layers than the VGG16	Used for recognition tasks of a large-scale number of images dataset
15.	Xception	An improved version of the Inception family of CNN	Used for classifying visual objects for computer vision applications with a slightly higher accuracy

TABLE 2: Parameters of the COVID-19 deep learning models.

Model no.	Deep learning model	Tuning parameters
1	CNN	Momentum = 0.5 to 0.9; number of epochs = 0.9; batch size = 32.
2	DarkNet	batch = 64; momentum = 0.9; learning_rate = 0.000008.
3	DNN	batch_size = $c$ (32, 64), dropout_rate = $c$ (0.1, 0.2, 0.3), units = $c$ (10, 20).
4	GoogleNet	Each one must be resized from $647 \times 511 \times 3$ to $227 \times 227 \times 3$ pixels, the dimensions used to train GoogleNet $224 \times 224 \times 3$ pixels.
5	InceptionResNetV2	Outputs = Dense (100, activation = 'softmax') (base_model.output) model = Model (base_model.inputs, outputs).
6	Inceptionv3	batch_size = $c$ 64, dropout_rate = $c$ (0.1, 0.2, 0.3), units = $c$ (10,20, 30).
7	LSTM	Rule search (evaluation measure) = entropy; minimum rule coverage = 2, maximum rule length = 6.
8	MobileNetV2	learning_rate = 0.0001; no. of epochs = 10.
9	NASNet-large	learning_rate = 0.0002; no. of epochs = 20.
10	ResNet34	Optimization method: Adam; momentum: 0.90; weight-decay: 0.0006; dropout: 0.6; batch size: 100; learning rate: 0.02; total no. of epochs: 20.
11	ResNet50	Optimization method: Adam; momentum: 0.97; weight-decay: 0.0005; dropout: 0.7; batch size: 100; learning rate: 0.03; total no. of epochs: 30.
12	SAE	batch_size = $c$ (64), dropout_rate = $c$ (0.1, 0.2, 0.4), units = $c$ (10, 20, 40).
13	VGG16	Optimization method: SGD; momentum: 0.90; weight-decay: 0.0004; dropout: 0.6; batch size: 164; learning rate: 0.06; total no. of epochs: 60.
14	VGG19	Optimization method: SGD; momentum: 0.97; weight-decay: 0.0005; dropout: 0.3; batch size: 128; learning rate: 0.07; total no. of epochs: 40.
15	Xception	Optimizer method: SGD; momentum: 0.8; learning rate: 0.035; learning rate decay: decay of rate 0.92 every 4 epochs.

on the ImageNet database that contains several millions of images [53]. It can transfer learning to classify objects of new untrained images. The network receives images of the 331-by-331 size and extracts features of objects such as a computer mouse, keyboard, pencil, animals, etc.

- (x) ResNet34: The ResNet34 is an improved version of the residual neural network (ResNet) base model with 34 layers. The ResNet has a CNN architecture but with a shortcut mechanism between layers to speed up solving problems. The shortcut mechanism prevents alteration and reduces the complexity of the network. Moreover, the ResNet model has a bottleneck blocks mechanism to speed up the network's training process [53].
- (xi) ResNet50: The ResNet50 is another version of the ResNet with 50 layers. It differs from the ResNet34 by trending to perform deeper analysis to solve complex problems, and hence it has better object classification and recognition accuracy. However, in many cases, a deeper analysis degrades the accuracy of the network. As with many other CNN models, the ResNet50 is tested using the ImageNet dataset to recognize and classify objects and shows high-quality performance [53].
- (xii) Stacked autoencoder: The stacked autoencoder (SAE) is a multilayer DNN that entails input, hidden, and output layers. The number of neurons in the input layer and output layer are equal, and the hidden layer usually has lesser neurons than them. The SAE has an unsupervised learning algorithm that differs from other networks by a stacked autoencoder [54]. The stacked autoencoder performs a coding and decoding process in which the coding takes place for the data of the input and the hidden layers, while decoding takes place for the data of the hidden and the output layers. The SAE provides a feature fusion mechanism that enables it to deal with a small dimension of features.
- (xiii) VGG16: The VGG16 is another type of CNN model (a variant of the main VGG model, including VGG11 and VGG19) that consists of 16 layers. Simonyan and Zisserman propose it for processing large-scale image recognition tasks. It differs from the AlexNet model by integrating multiple  $3 \times 3$  kernel-sized filters instead of kernel-sized filters in the convolutional layers. It produces considerably high accuracy results than some other popular models but with the cost of longer training time. The VGG16 model achieves an accuracy of 92.7% when applied to the ImageNet challenges and scores a place in the top 5 CNN models [55].
- (xiv) VGG19: The VGG19 is another variant of the main VGG model, which consists of 16 convolution layers, 5 MaxPool layers, 3 fully connected layers, and 1 SoftMax layer (i.e., 19 layers in total). The

input to the VGG19 is an RGB image of a fixed size ( $224 * 224$ ) that is represented by a  $(224, 224, 3)$  matrix [56].

- (xv) Xception: The Xception is a recent CNN model that is inspired by the InceptionResNet [39]. The InceptionResNet, Inceptionv3, and later Xception are characterized by the addition of the Inception module, but its have different versions of various parameters [18]. The improvements of the Xception enable it to slightly outperform the older version when applied to the ImageNet dataset.

*3.1.3. Data Preprocessing.* In the preprocessing stage of the COVID-19 images, CNN models use annotating lesions for each CT volume in which lung masks are implemented in the training phase to form a mask volume. This mask volume is concatenated with the CT volume to obtain the final CT volume. The final CT volume is then set to a specific resolution (e.g.,  $200 * 400$ ) to prepare it for DL execution. The number of slices of the image sample is fixed and does not change during the image preprocessing phase (they have a range of 73 to 250 in the testing dataset). In this research, the images of the COVID-19 are collected from different sources, including imaging clinics and existing datasets. The images are captured by different types of equipment and contain different acquisition parameters. As a result, there exist considerable variations in the intensity of the images [18]. However, the proposed CNN models implement two standard preprocessing procedures of resizing and normalization to ensure that the CNN models' generalization is not negatively affected.

- (i) Resizing: we need first to acquire a constant dimension because all images in this dataset vary in dimension and resolution ( $365 * 465$  to  $1125 * 859$  pixels). Subsequently, all the images are scaled to specific pixels based on the corresponding CNN models (e.g., NASNet-Large  $331 * 331$  pixels and NASNetMobile  $224 * 224$  pixels).
- (ii) Normalization: In the normalization part, to set the scaling limit, we use a precalculated mean subtraction of the ImageNet database to normalize the intensity values [56]. Then we scale the intensity values from  $[0, 255]$  to the intensity range of  $[0, 1]$  using the min-max normalization formula.

$$x_{\text{norm}} = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}. \quad (1)$$

*3.1.4. Deep Learning for Feature Extraction.* The quality and quantity of the extracted features from images play an important role in producing robust and accurate diagnosis results. The DL CNN requires a much larger number of features and data size to minimize the error of the classification. A small-scale data might cause problems such as imbalanced training and overfitting [57]. The feature extraction of this work is first implicitly performed according to the CNN models' standard architecture, as each model has

its own existing feature extraction algorithms. However, to obtain more significant classification results, we need to increase the dimensionality of the features as the COVID-19 dataset considerably has a small number of images. Hence, we implement selected feature extraction algorithms that are suitable for deep convolutional feature representation to produce an additional computed encoded feature vector [58]. The initially computed feature vector of the CNN and the computed encoded feature vector are combined in one final feature vector. The corresponding classifiers use the final feature vector to produce the diagnosis results.

**3.1.5. Evaluation Measurements.** In this study, we used the most common evaluation metrics, including the area under the curve (AUC), classification accuracy rate (CAR),  $F1$ -score, receiver operating characteristics (ROC) curve, precision, recall, and mean squared error (MSE). They are calculated based on the classification results of true positive rate (TPR), false positive rate (FPR), predictive positive value (PPV), and negative predictive value (NPV). Accordingly, equations (2) to (6) represent the mathematical description of the evaluation metrics:

- (i) Classification Accuracy Rate: It is known as classification accuracy rate (CAR), and the CAR shows how the output results near for the actual outcomes, which is calculated using the following equation:

$$CAR = \frac{(TP + TN)}{(TP + TN + FP + FN)} \times 100. \quad (2)$$

- (ii) Precision: It is utilized to measure the ability of a classifier to identify the importance of classified subjects and reject insignificant subjects, which is calculated using the following equation:

$$\text{Precision} = \frac{TP}{TP + FP}. \quad (3)$$

- (iii) Recall: It is utilized to measure and evaluate the importance level of the classified subjects, which is calculated using the following equation:

$$\text{Recall} = \frac{TP}{TP + FN}. \quad (4)$$

- (iv) ROC: The ROC curve is utilized to graphically plot the classifier's overall performance in terms of providing the correct results by plotting the TPR against the FPR.

- (v) Mean Absolute Error: The mean absolute error (MAE) is a linear score that is widely used for calculating classification error, which is calculated using the following equation:

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|. \quad (5)$$

- (vi)  $F1$ -score: The  $F1$ -score is extracted from measuring the precision and recall in which the best  $F1$ -score

has the value of 1 or near to the value of 1 and the worst  $F1$ -score has the value of 0 or near to the value of 0, which is calculated using the following equation:

$$F1 - \text{score} = \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (6)$$

**3.2. Weighting Stage.** The crow swarm optimization (CSO) is used to find the best set of coefficients that can be applied through a designed fitness function to evaluate the performance of the used deep learning algorithms. CSO is designed in a way that will ensure a good distribution for all the selected coefficients, this was done by taking the best available average fitness (not just the best one). In this section, the inspiration of the proposed method and the mathematical model are discussed.

**3.2.1. Inspiration.** The American crow (*Corvus brachyrhynchos*) is an example of a species that has evolved complicated social behaviors. They are kind of crows, conjointly called the common crow. They are living in North-western yank. The crows are divided into teams, in which the typical cluster size of American crows outbound, their last hunt site of the day was  $237 \pm 43$ . They are ready to reach speeds of 35–43 mph. Daytime hunt aggregations of crows throughout the nonbreeding season area unit sometimes composed of various family teams [59]. Yank Crow's kind communal roosts will vary from 100 to 2 million crows [59]. However, these hunt sites' area unit are usually not among the visibility of their roosting sites. American crows' area unit make a superb model species to conduct analysis on social behavior because they prominently forage in teams before sunset and form massive communal roosts in the dark throughout the nonbreeding season [60].

Communal roosts perform as info-sharing centers. At these communal roosts, crows share info like wherever to forage throughout the day. Crows that have not found sensible an honest forage site can follow crows that have found good forage sites the following day [59]. Communal roosts conjointly aid in thermoregulation and predator turning away [42]. The dimensions of the communal roost tend to extend because the weather gets colder. In distinction to their evening communal roosts, American crows are divided into smaller teams to forage throughout the day. Each morning, crows disperse from their communal roosting site and travel up to 40 miles away to forage.

These three predictions that will facilitate support this hypothesis:

- (i) American crows can leave their communal search sites in giant teams, as critical singly.
- (ii) American crows can depart their communal forage websites when the sunset and fly along towards their communal roosting site to own the longest forage opportunities day by day.
- (iii) American crows can fly within the direction of their communal roosting site once outward their last communal search site.



3.2.2. *Mathematical Model and Algorithm.* The algorithm of CSO together with its mathematical model is presented in these subsections.

(1) *Group Division.* The CSO algorithmic program mimics the behavior of *Corvus brachyrhynchos*. To model such interactions, every cluster of crows' area unit needed to maneuver over the search area. As mentioned earlier, the crows being divided into teams who begin to look for places of food at long distances area and not among the scope of traditional vision. Assume the crow's algorithmic program determines the simplest cluster you get when choosing the food space and additionally deciding the totals that did not get sensible food on this trip. Within the next journey of food search, teams with dangerous food can eat sensible food. Reckoning on the characteristics of the animal, like speed, angle for departure, and placement. This behavior is delineating by the following equation:

$$\mathbf{S}_{i+1} = (\mathbf{R} * \mathbf{S}_i) + (\mathbf{P}_{\text{best}} - \mathbf{P}_i) * \mathbf{R}, \quad (7)$$

where  $i$  indicates the current iteration,  $R$  is a random number between (0,1),  $\mathbf{P}_{\text{best}}$  is the position vector of the best solution obtained so far,  $P_i$  is the position vector,  $S_i$  is the velocity value, the velocity is between (-6, 6).

(2) *Update Position.* The update position for each crow in the group depends on the best crow's position in that group, and this can be done using the following equation:

$$\mathbf{P}_{i+1} = \mathbf{P}_i + \mathbf{S}_{i+1} * \cos(\theta), \quad (8)$$

where  $P_{i+1}$  is a new position and  $\Theta$  is an angle for departure between (45, 135). The position should be updated in each iteration.

(3) *Update the Angle.* When the best crow follows the behavior, the worst crow angle will be updated according to the following equation:

$$\frac{\theta_{i+1} + 1 = (\theta_i + \theta_{\text{best}})}{2}, \quad (9)$$

where  $\Theta_{i+1}$  is the new angle for crow,  $R$  is a random number between (0, 1),  $\Theta_i$  is the current angle of crow, and  $\Theta_{\text{best}}$  is the angle of best crow.

The important stage in the operations of a solution is the initialization process that provides the algorithm needs and the data of the problem and submits it. The preparedness phase consists of several stages. The first phase is the process of reading the problem database information. Thereafter generate several units for all measures. The sum of units must be 100 that are randomly distributed for the nine measures.

In the second stage, we applied the CSO algorithm to solve this problem. It is started by calculating the value of the initial speed and the angle for all crows. In this step, the speed value is between (-6, 6), and the angle value is between (45, 135). This procedure mimics the situation in a real American crow. This case represents the crow's first movement to search for the source of the feed (food). When

selecting any path, crows can receive quantities of food. These routes do not necessarily lead to the feed source. Therefore, the food during this case is a guide that works on the ways that are taken by crows and not necessarily the food path. A solution can be constructed using speed, angle, and best position in-group. After the initialization step, the CSO algorithm starts to work. Crows start to move from the beginning node that had been chosen in the initialization stage. The following fitness function is used to evaluate each solution, the problem is formulated as a 2D matrix (15 \* 9), as there are 15 algorithms to be evaluated with nine measures, and the goal of the proposed fitness equation is to find the overall performance for each one of 15 algorithms.

$$\begin{aligned} \text{Fitnees}(i) = & x1 * \text{AUC}(i) + x2 * \text{CAR}(i) + x3 * \text{FScore}(i) \\ & + x4 * \text{Precision}(i) + x5 * \text{Recall}(i) \\ & + -x6 * \text{FPR}(i) + x7 * \text{PPV}(i) + x8 * \text{NPV}(i) \\ & - x9 * \text{MSE}(i). \end{aligned} \quad (10)$$

We update position (number of units) depending on the previous position, speed, and crow angle. Before updating the position, the speed must be updated depending on the previous speed, which is different between the positions of the best crow and the current crow. The third stage is the updated angle for every crow. Lastly, after several iterations, we returned the best average for all students and distributed the units for the classifier. Pseudocode of CSO is presented in Algorithm 2.

where  $i$  indicates the current iteration,  $R$  is a random number between (0, 1),  $P_{\text{best}}$  is the position vector of the best solution obtained so far,  $P_i$  is the position vector,  $S_i$  is the velocity value, the velocity is between (-6, 6).  $P_{i+1}$  is a new position, and  $\Theta$  is an angle for departure between (45,135).  $\Theta_{i+1}$  is the new angle for crow,  $R$  is a random number between (0, 1),  $\Theta_i$  is the current angle of crow, and  $\Theta_{\text{best}}$  is the best crow angle. The position should be updated in each iteration.

3.2.3. *CSO Benchmarking with Other Swarm Optimization Algorithms.* CSO algorithm is evaluated by using 30 benchmark functions. Some of those functions are standard functions that are used in researches. These functions are chosen to be able to show the performance of CSO and to compare it with some known algorithms. The selected 30 test functions are shown in Tables 3 and 4, where  $D$  means the function's dimension, range means the function's search space limits, and Opt is the optimal value. The selected functions are unimodal or multimodal benchmark minimization functions. Unimodal test functions have a single optimum value; thus, they can benchmark an algorithm's convergence and exploitation. Multimodal test functions have more than one optimum value, making them more challenging than unimodal. An algorithm should avoid all the local optima to approach and approximate the global optimum. So, exploration and local optima avoidance of algorithms can be benchmarked by multimodal test functions.

- (1) Maximum\_number\_of\_iterations = 1000.
- (2) Initialize population of 15 solutions, and each solution is of length 11 (9 for the metrics and the angle and speed).
- (3) Initialize speed  $S$  and angle  $\Theta$ , where ( $S \in [-6, 6]$ ), ( $\Theta \in [45, 135]$ ).
- (4) Calculate the fitness (general average) for all solutions, using equation (10)
- (5) Select the best value of the general average and store it in CurrentBest.
- (5)  $t = 1$ .
- (6) While ( $t \leq \text{maximum\_number\_of\_iterations}$ )
- (7) Update the position of all solutions, using the following:  $S_{i+1} = (R * S_i) + (P_{\text{best}} - P_i) * RP_{i+1} = P_i + S_{i+1} * \cos(\theta)$
- (8) Update the fitness of all solutions.
- (9) Select the best value of fitness and store it in NewBest.
- (10) Update CurrentBest:  
if NewBest is better than CurrentBest then CurrentBest = NewBest.
- (11) Update the angle of solutions using the following:  
 $\Theta_i + 1 = (\Theta_i + \Theta_{\text{best}}) / 2$
- (12)  $t = t + 1$ .
- (13) End while.
- (14) Return the solution that has the best fitness.

ALGORITHM 2: Pseudocode of CSO.

TABLE 3: Unimodal benchmark functions.

Function	Equation	Test name	$D$	Range	Opt
F1	$f_1(x) = \sum_{i=1}^n x_i^2$	Sphere	30	-100, 100	0
F2	$f_2(x) = \sum_{i=1}^n  x_i  + \prod_{i=1}^n  x_i $	Schwefel 2.22	2	-100, 100	0
F3	$f_3(x) = \max_i \{ x_i , 1 \leq i \leq n\}$	Schwefel 2.21	2	-100, 100	0
F4	$f_{31}(x) = 2x_1^2 - 1.05x_1^4 + x_1^6/6 + x_1x_2 + x_2^2$	Three-Hump Camel	2	-5, 5	0
F5	$f_6(x, y) = -200e^{-0.5\sqrt{x^2 + y^2}}$	Ackley 2	2	-32, 32	-200
F6	$f_7(x) = x_1^2 + x_2^2 - 0.3 \cos(3\pi x_1) - 0.4 \cos(4\pi x_2) + 0.7$	Bohachevskyn N.1	2	-100, 100	0
F7	$f_8(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$	Booth	2	-10, 10	0
F8	$f_{38}(x) = -\sum_{i=1}^d (x_i - 1)^2 - \sum_{i=2}^d x_i x_{i-1}$	Trid	6	-36, 36	-50
F9	$f_9(x) = \sum_{i=1}^n x_i^2 + (\sum_{i=1}^n 0.5ix_i)^2 + (\sum_{i=1}^n 0.5ix_i)^4$	Zakharov	2	-5.12, 5.12	0
F10	$f(x) - 1 + \cos(\sqrt{x_1^2 + x_2^2}) / 0.5(x_1^2 + x_2^2) + 2$	Drop Wave	2	-4.5, 4.5	-1
F11	$f_{13}(x) = \sum_{i=1}^n x_i^{10}$	Schwefel 2.23	2	-100, 100	0
F12	$f_{14}(x) = \sum_{i=1}^n  x_i $	Schwefel 2.20	2	-100, 100	0
F13	$f_{18}(x) = \sum_{i=1}^{d/4} [(x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} + x_{4i})^2 + (x_{4i-2} + x_{4i-1})^4 + 10(x_{4i-3} + x_{4i})^4]$	Powell	10	-4, 5	0
F14	$f_{19}(x) = \sum_{i=1}^d [(\sum_{j=1}^d x_j^i) b_i]^2$	PowerSum	4	0, 4	0
F15	$f_4(x) = \sum_{i=1}^n [b(x_{i+1} - x_i^2)^2 + (a - x_i)^2]$	Rosenbrock	30	-2.048, 2.048	0

For each benchmark function, the CSO algorithm and the compared algorithms are performed in the experiments under the condition of the same number of iterations (1000), independent runs for 30 times, and the population size is set to 50. The statistical results (average and standard deviation) are shown in Tables 5 and 6. For verifying the results, the CSO algorithm is compared with GWO [61], HHO [62], SSA [63], and WOA [64].

The results in Table 5 demonstrated that CSO is better than the selected algorithms in most unimodal (nine out of 15) test functions. Unimodal functions test the exploitation of an algorithm. The obtained results showed CSO

superiority in exploiting the optimal value, so CSO provides excellent exploitation ability.

For testing the exploration strength of an algorithm, the multimodal functions are used as the number growing exponentially with dimension such types of functions. The results in Table 6 demonstrated that CSO is better than the selected algorithms on most (13 out of 15) multimodal functions. The obtained results show the superiority of the CSO algorithm in terms of exploration. Algorithms' average fitness on the test functions is presented in Figure 3, and the standard deviations of the algorithms on the test functions are shown in Figure 4.

TABLE 4: Multimodal benchmark functions.

Function	Equation	Type	Test name	D	Range	Opt
F16	$f_{20}(x) = -20 \exp(-\sqrt{0.2} \sum_{i=1}^n x_i^2) - \exp(1/n \sum_{i=1}^n \cos(2\pi x_i)) + 20 + e$	N	Ackley	2	-10, 10	0
F17	$f_{22}(x) = \sum_{i=1}^n i x_i^4 + \text{random0}, 1$	N	Quartic	10	-1.28, 1.28	0+rand
F18	$f_{23}(x) = (4-2.1 x_1^2 + x_1^4/3) x_1^2 + x_1 x_2 + (-4+4 x_2^2) x_2^2$	N	Six-Hump Camel	2	-5, 5	-1.0316
F19	$f_{24}(x) = a(x_2 - b x_1^2 + c x_1 - r)^2 + s(1-t) \cos(x_1) + s$	N	Branin	2	-5, 15	0.3979
F20	$f_{25}(x) = [1 + (x_1 + x_2 + 1)^2 (19 - 14 x_1 + 3 x_1^2 - 14 x_2 + 6 x_1 x_2 + 3 x_2^2)] * [30 + (2 x_1 + 3 x_2)^2 (18 - 32 x_1 + 12 x_1^2 - 48 x_2 + 36 x_1 x_2 + 27 x_2^2)] *$	N	Goldstein Price	2	-2, 2	3
F21	$f_{26}(x) = -\sum_{i=1}^4 c_i \exp(-\sum_{j=1}^3 a_{ij} (x_j - p_{ij})^2)$	F	Hartmann 3-D	3	1, 0	-3.8628
F22	$f_{27}(x) = -\sum_{i=1}^4 c_i \exp(-\sum_{j=1}^6 a_{ij} (x_j - p_{ij})^2)$	F	Hartmann 6-D	6	1, 0	-3.3224
F23	$f_{32}(x) = -200 e^{-0.2 \sqrt{x^2 + y^2}} + 5 e^{\cos(3x) + \sin(3y)}$	N	Ackley 3	2	-32, 32	-195.629
F24	$f_{33}(x) = x_1^2 + 2x_2^2 - 0.3 \cos(3\pi x_1) \cos(4\pi x_2) + 0.3$	N	Bohachevskyn N.2	2	-10, 10	0
F25	$f_{34}(x) = \sin(x) e^{(1-\cos(y))^2} + \cos(x) e^{(1-\sin(x))^2} + (x-y)^2$	N	Brid	2	-2pi, 2pi	-106.7645
F26	$f_{35}(x) = ( \sin(x_1) \sin(x_2) \exp( 100 - \sqrt{x_1^2 + x_2^2}/\pi  + 1) )^{0.1}$	N	Cross in Tiny	2	-10, 10	-2.06261
F27	$f_{36}(x) = -\cos(x_1) \cos(x_2) \exp(-(x_1 - \pi)^2 - (x_2 - \pi)^2)$	F	Easom	2	-100, 100	-1
F28	$f_{37}(x) = -\sin^2(x-y) \sin^2(x+y) / \sqrt{(x^2 + y^2)}$	N	Keane	2	0, 10	-0.6737
F29	$f_{41}(x) = - \sin(x_1) \exp( 1 - \sqrt{x_1^2 + x_2^2}/\pi ) $	N	Holder	2	-10, 10	-19.2085
F30	$f_{43}(x) = -\sum_{i=1}^d \sin(x_i) \sin^{2m}(i x_i^2/\pi)$	N	Michalewics	2	1.57, 2.21	-1.8013

TABLE 5: Unimodal benchmark functions.

Name	CSO		GWO		HHO	
	AV	STD	AV	STD	AV	STD
Sphere	1.10132E-12	2.17067E-12	8.32056E-62	2.01043E-61	5.75505E-96	1.21415E-95
Schwefel 2.22	5.5237E-112	1.5496E-111	8.74E-99	0	1.11321E-48	1.51435E-48
Schwefel 2.21	1.4406E-111	6.7475E-111	1.26E-105	0	2.95204E-51	6.83343E-51
Camel3	0	0	0	0	1.1048E-107	2.3653E-107
Ackley2	-200	0	-200	0	-200	0
Bohachevskyn N. 1	0	0	0	0	0	0
Booth	0	0	1.54098E-07	1.11392E-07	2.36488E-05	2.5515E-05
Trid	-50	0	-49.99989	8.03012E-05	-1367.28225	4.374233259
Zakharov	7.9142E-221	0	0	0	9.5294E-47	2.13084E-46
Drop Wave	-0.98496333	0.027373716	-0.99574666	0.016186579	-1	0
Schwefel 2.23	0	0	0	0	0	0
Schwefel 2.20	1.0876E-111	5.4846E-111	3.21E-88	0	3.48466E-47	4.72514E-47
Powell	0.001705466	0.001128045	4.94865E-07	6.09827E-07	4.9489E-104	7.662E-104
PowerSum	0.051633374	0.070129514	0.106802381	0.261322987	1.4218E-131	2.4194E-131
Rosenbrock	7.610173333	21.01106746	26.74104667	0.704408086	0.008242153	0.010289518

3.3. *Selection Stage.* In this stage, among 15 deep learning models, the final diagnostic model will be selected based on evaluation experiment with CSO benchmarked algorithm. Furthermore, the winner model is the one that achieved best results in all evaluation measurements without presenting any overfitting or underfitting classification performance in all classes.

## 4. Results and Discussion

4.1. *Results of Evaluation Stage.* To the best of the authors' knowledge, this was the first study to carry out supervised COVID-19 identification with large numbers of CT cases within the cutting-edge healing center during the current COVID-19 pandemic through developing and training a

successful deep learning model on collected Lung CT samples. The study's motivation is to utilize AI to lighten the issue of proficient interpretation deficiencies for CT lung scan when the virus is still spreading rapidly. Although there were numerous viable AI uses in past works [7], improving AI for automated COVID-19 identification is still challenging. First, the number of patients registered is moderately littler within the current crisis circumstance than in past works [2, 27]. The patients selected in our research are clinically analyzed with COVID-19 as most of the patients did not experience PCR testing due to an unexpected outbreak and limited therapeutic resources in a limited time. Second, radiologists did not label COVID-19 samples in CT lung scan regions and labeled COVID-19 samples for the patients can be classified (COVID-19 or

TABLE 6: Multimodal benchmark functions.

Name	CSO		GWO		HHO	
	AV	STD	AV	STD	AV	STD
Ackley	$8.8818E-16$	$4.01173E-31$	$8.8818E-16$	$4.01173E-31$	$-8.8818E-16$	0
Quartic	0.031253333	0.018374115	0.000971688	0.000902519	0.000445553	0.000297211
6-Hump Camel	-1.0316	$6.77522E-16$	-1.031628448	$4.31084E-09$	-1.0316	0
Branin	0.3979	0	0.397887852	$5.97628E-07$	0.397946667	$9.81495E-05$
Goldstein	3	0	3.000007881	$9.28286E-06$	3	0
Hart3	-3.8628	$3.16177E-15$	-3.8620812	0.001938044	-3.854683333	0.008125864
Hert6	-3.3224	$1.35504E-15$	-3.264253912	0.099876517	-2.9311	0.07435459
Ackley3	-195.629	$5.78152E-14$	-195.6290282	$2.8506E-08$	-186.4112	0
Bohachevskyn N. 2	0	0	0	0	0	0
Bird	-106.7645	$7.2269E-14$	-106.1160683	3.551734418	-106.7645	-106.7645
Cross_in_Tiny	-2.06261	0	-2.062611869	$3.30268E-09$	-2.0626	0
Easom	-1	0	-1	0	-0.99998	$8.16497E-06$
Keane	-0.6737	$1.1292E-16$	-0.6737	$1.1292E-16$	-0.67367	0
Holder	-19.2085	$3.61345E-15$	-19.20849251	$8.17787E-06$	-19.2085	0
Michalewics	-1.8013	$6.77522E-16$	-1.8013	$6.77522E-16$	-1.8013	0

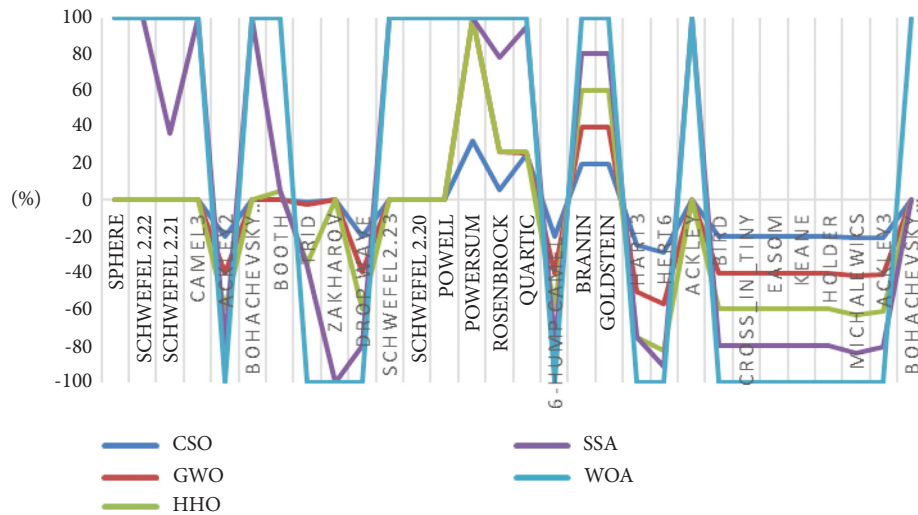


FIGURE 3: The average fitness of the algorithms on the test functions.

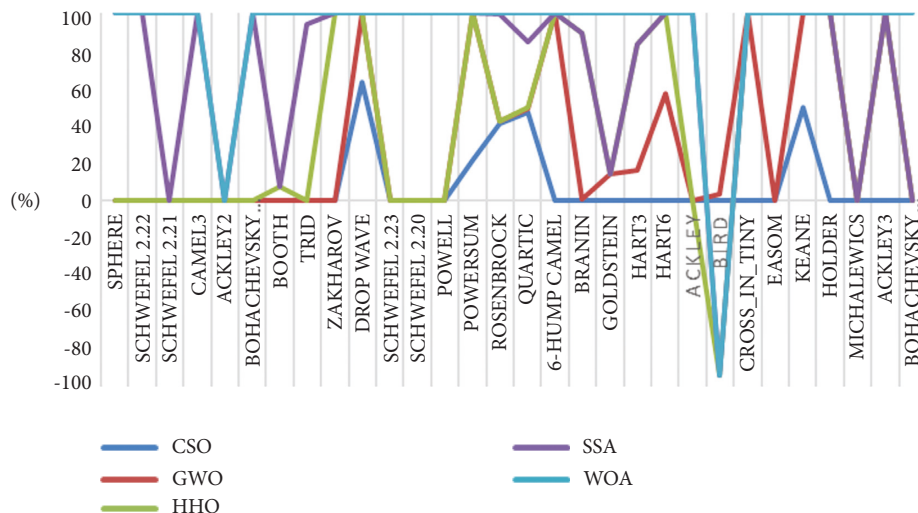


FIGURE 4: The standard deviations of the algorithms on the test functions.

healthy) to train the deep learning models in this research. Third, the proficient radiologist may miss a few tiny infected regions of COVID-19, and it remains unclear whether deep learning-based models can be detected. We hypothesized solving these problems by offering delicate deep learning models, that is, ResNet50, GoogleNet, and MobileNetV2. The first problem is solved using broad CT lung training samples to get perfect training accuracy. Considering the COVID-19 pandemic, the supervised learning issue is used for the second problem [29, 43], such as COVID-19 detection without explaining COVID-19 lesion areas. In this study, the temporally global pooling layer and spatially global pooling layer are utilized in the ResNet50 model to technically handle the COVID-19 identification issue. Finally, to address the third problem by taking advantage of deep learning models and using pretrained CNN models to provide lung masks to guide ResNet50, GoogleNet, and MobileNetV2 learning.

Diagnostic imaging techniques, such as chest radiography and CT, play a significant role in verifying the primary diagnosis of the polymerase chain reaction (PCR) test for COVID-19. Clinical imaging frequently plays a vital role in tracking disease development and patient treatment. Extracting features from radiology modes is an important step in developing profound education models as model success depends directly on the features. This research aims to present an exhaustive study on the classification of COVID-19 in CT imaging using state-of-the-art deep CNN architectures extracted from features and trained in machine learning algorithms based on deep learning models' successful computer vision. The 3-fold cross-validation technology was developed to evaluate each experiment's average classifier generalization performance. For all CNNs, network weights from the weights trained in ImageNet have been initialized. The computer system used in this project, based on Windows, included an Intel(R) Core (TM) i7-8700K 3.7 GHz processor with 32 GB RAM. Python has introduced the training and testing phase of the proposed architecture using a Tens row backend kit as a deep learning application backend, using an 11 GB RAM, NVidia GeForce GTX 1080 Ti GPU.

Popular pretrained and typical models such as ResNet50, DarkNet, GoogleNet, MobileNetV2, Xception, VGG19, VGG16, InceptionV3, ResNet34, CNNs, DNN, SAE, InceptionResNetV2, LSTM, and NASNet-Large models have been trained and tested on CT lung images. The training accuracy and loss values for fold-3 of the pretrained models are shown in Figures 5 and 6, respectively.

The training stage was carried out until the 30th epoch to prevent overfitting for all pretrained models. The diagnostic model's performance outcomes from various pretrained CNN and deep models are in Table 7.

The COVID-19 diagnostic-based deep learning models used in our research are successful compared to recent deep learning computer-aided diagnostic approaches. The deep learning models for 746 CT cases were trained to predict the COVID-19 risks and its influence for early prediction [13]. The deep learning model of 746 lung CT scans was trained for essential lung CT findings [23], and the ROC AUC metric was obtained of 0.92. In our analysis, the 522 scans only are utilized for training tasks; on the other hand, the

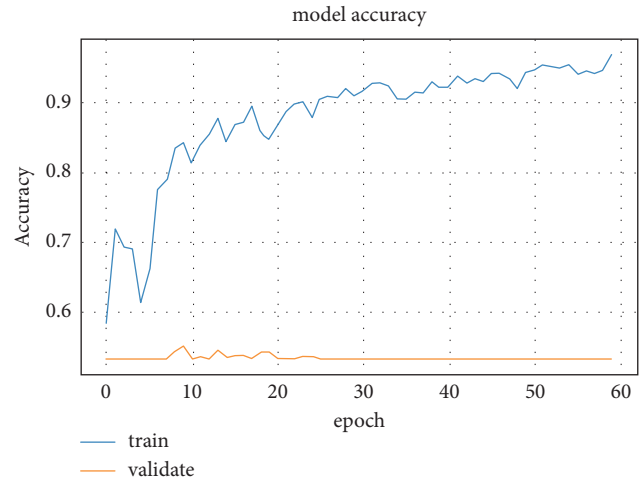


FIGURE 5: First dataset training accuracy for ResNet50 model.

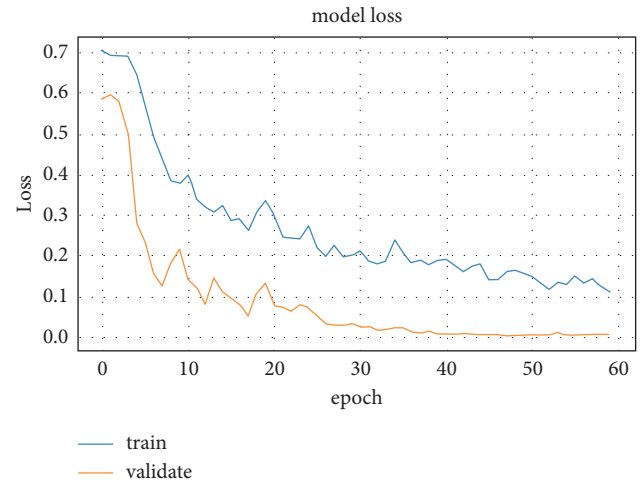


FIGURE 6: First dataset loss values for ResNet50 model.

ROC AUC metric is obtained of 0.90. Based on the variation in the dataset and compared with deep learning models, it was conceivable to identify that classifying COVID-19 can be more straightforward, and the proposed deep learning models are exceptionally effective. As for the incorrect 12 false negative predicted cases, after rechecking the original CT lung cases, the reasons are as follows: there is a slight increase in CT lung regions, and the CT cases of this ground-glass ambiguity are fragile without integration process.

Our research aims to provide a good and promising solution for improving medical diagnostic approaches based on AI for urgent diseases like the COVID-19 pandemic. In addition, to improve the identification, the approach is utilized as a helpful instrument to assist the specialists within the health and medical centers to choose which ideal deep learning model could be used for COVID-19 identification by assessing distinctive deep learning models. The COVID-19 diagnostic deep learning-based models entail the CT lung scan within the current austere battle against this pandemic. However, doctors in hospitals and medical centers are busy

TABLE 7: The first dataset diagnostic performance outcomes of various pretrained models.

No	Classifier	AUC	CAR	<i>F1</i> -score	Precision	Recall	FPR	PPV	NPV	MSE
1	ResNet50	90.78	91.46	90.49	89.73	88.94	90.92	90.22	90.17	0.039
2	DarkNet	80.92	85.13	85.11	83.29	80.14	82.19	83.71	81.95	0.069
3	GoogleNet	86.99	90.35	89.42	90.21	90.11	88.92	90.47	90.16	0.044
4	MobileNetV2	85.47	88.36	87.69	83.33	86.91	82.14	85.25	86.94	0.038
5	Xception	75.14	77.13	75.02	76.96	74.82	77.37	74.38	72.96	0.094
6	VGG19	80.32	84.38	84.29	77.43	93.56	82.14	83.64	81.79	0.081
7	VGG16	81.36	80.14	80.25	78.97	79.46	78.19	80.03	76.91	0.078
8	InceptionV3	63.40	65.98	64.81	66.47	64.91	62.99	65.11	63.73	0.055
9	ResNet34	90.54	90.71	80.48	79.28	80.10	89.79	90.21	89.94	0.042
10	CNNs	87.47	88.15	83.36	87.55	87.89	86.36	87.99	86.69	0.057
11	DNN	83.39	85.36	81.30	83.57	85.66	83.96	84.14	82.64	0.063
12	SAE	80.39	82.14	79.92	84.87	81.93	80.97	83.94	83.12	0.059
13	InceptionResNetV2	85.67	87.95	86.11	87.64	84.24	86.14	88.19	85.37	0.098
14	LSTM	88.25	90.54	88.36	89.25	88.97	86.91	90.11	87.67	0.096
15	NASNet-Large	79.36	80.11	78.98	77.91	78.16	75.87	79.47	82.96	0.079

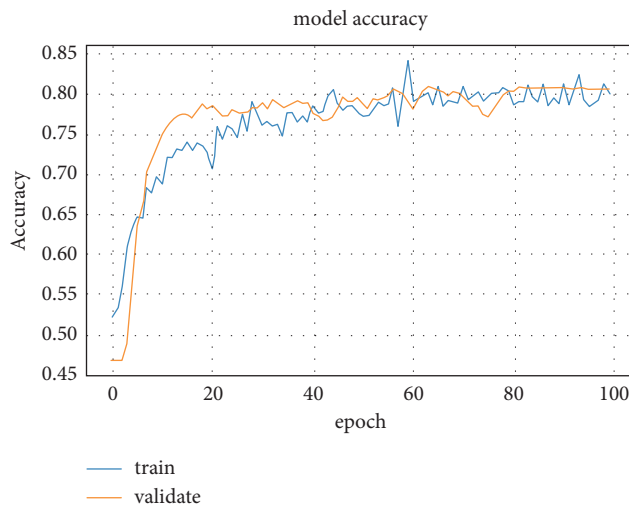


FIGURE 7: Second dataset training accuracy for ResNet50 model.

treating many COVID-19 patients, and it may be hard for them to do the CT scan for all patients. In this study, the supervised deep learning method provides the location of pulmonary in CT lung lesions.

The annotation attempts of radiologists can be reduced, such as just the provision of COVID-19 classification as a healthy or COVID-19 case. As a result, developing a useful AI instrument has quickly become potential and accessible in clinical applications. In the future, automatic deep learning could significantly reduce the burden on AI experts. The training accuracy and loss values for fold-3 of the pretrained models are shown in Figures 7 and 8, respectively.

The training stage was carried out until the 30th epoch to prevent overfitting for all pretrained models. The diagnostic model's performance outcomes from various pretrained CNN and deep models are presented in Table 8.

**4.2. Results of Weighting Stage.** The COVID-19 optimization method has two significant benefits over other related approaches. Firstly, the input parameters were known defined



FIGURE 8: Second dataset loss values for ResNet50 model.

by disease statistics, stopping scientists from initializing them with random values. Second, the approach can end after several iterations without setting this value. Infected populations initially increase exponentially, but the number of infected people declines after some iterations. The algorithm is executed ten times, and the average results are taken to avoid any randomness in taking the best result only. The ten runs and their average results are shown in Table 9 for the first and second datasets. Each value in the last row of each dataset represents the average of the ten results of each measure. The fitness function uses these averages to get the final performance for each algorithm.

Subsequently, the final result column in the first dataset of Table 10 is obtained by applying equation (10) (fitness function). The values of  $x_1$  to  $x_9$  are the average values taken from the last row in Table 9. The average values for the 15 algorithms are shown in Table 10 for the first and second datasets, respectively. The final result represents each deep learning algorithm's overall strength (assessment). The results of the first dataset in Table 10 showed that the ResNet50 algorithm is the best-selected algorithm, and it is of close performance to GoogleNet and LSTM algorithms.

TABLE 8: The first dataset diagnostic performance outcomes of various pretrained models.

No	Classifier	AUC	CAR	<i>F1</i> -score	Precision	Recall	FPR	PPV	NPV	MSE
1	ResNet50	90.89	89.10	87.23	88.63	87.74	88.93	88.75	90.33	0.054
2	DarkNet	82.22	84.23	83.45	82.27	82.34	84.16	83.21	82.44	0.075
3	GoogleNet	84.76	86.76	84.91	85.20	85.10	85.72	83.33	84.36	0.059
4	MobileNetV2	83.12	85.33	84.19	84.37	83.49	84.87	85.10	83.22	0.047
5	Xception	84.83	81.35	80.12	79.99	80.43	80.84	80.32	80.98	0.061
6	VGG19	84.22	84.97	84.77	82.46	82.86	83.33	83.47	81.23	0.046
7	VGG16	91.34	89.96	88.75	88.15	88.95	88.97	89.14	87.99	0.038
8	InceptionV3	72.21	75.34	73.81	73.47	73.83	74.87	72.34	74.84	0.069
9	ResNet34	90.42	88.21	86.48	82.97	87.22	87.53	87.39	76.27	0.048
10	CNNs	85.29	83.65	83.10	82.55	81.76	82.40	82.77	82.74	0.050
11	DNN	81.96	83.50	82.37	83.12	82.12	83.12	81.94	82.84	0.083
12	SAE	83.10	82.90	81.87	80.43	81.27	81.84	81.36	81.38	0.078
13	InceptionResNetV2	88.38	88.35	87.43	86.76	86.42	87.53	88.05	87.46	0.069
14	LSTM	82.44	83.24	82.33	82.58	83.09	82.77	82.55	79.24	0.099
15	NASNet-Large	81.11	82.35	80.48	80.21	80.68	81.31	81.68	82.16	0.062

TABLE 9: CSO results for 10 runs.

Fitness	AUC	CAR	<i>F1</i> -score	Precision	Recall	FPR	PPV	NPV	MSE	Speed	Angle
<i>The results of the first dataset</i>											
71.82	14.45	17.46	10.59	8.50	10.16	9.63	5.72	5.19	18.28	0.69	53
81.32	14.76	12.38	17.45	9.11	10.53	6.66	5.77	3.49	19.81	0.77	129
77.12	14.26	18.53	13.51	7.51	9.96	8.46	6.05	3.42	18.27	-3.43	53
70.41	13.72	16.09	9.28	7.80	9.46	9.49	6.76	5.36	22.02	-2.22	45
77.73	14.30	18.10	11.57	12.52	6.01	6.87	4.06	4.13	22.42	-2.08	131
72.54	14.32	18.19	8.71	12.45	11.10	8.41	4.04	4.16	18.58	3.89	59
73.34	13.85	11.17	10.51	10.37	8.70	8.18	6.76	3.92	26.51	-3.31	57
73.57	15.19	11.23	10.65	9.18	9.31	7.60	8.09	6.78	21.94	3.99	67
78.56	14.73	17.97	9.035	13.52	10.27	5.60	6.08	3.17	19.60	2.46	57
76.20	15.55	12.41	11.11	13.38	7.11	7.33	6.25	3.65	23.17	-2.92	48
75.26	14.51	15.35	11.24	10.43	9.26	7.82	5.96	4.33	21.06	-0.21	69.9
<i>The results of the second dataset</i>											
74.26	14.18	14.50	12.00	9.93	10.21	8.70	5.11	5.50	19.85	3.71	57
79.03	13.89	12.48	13.01	14.99	9.42	5.79	4.54	6.00	19.86	2.08	46
74.31	14.15	18.68	12.20	11.80	7.76	7.64	5.38	5.44	16.93	-5.67	112
74.45	14.02	18.21	10.04	10.04	6.37	7.35	4.50	4.77	24.68	2.42	110
80.88	14.04	17.33	15.03	12.88	8.79	6.03	3.88	4.60	17.41	-0.02	56
81.85	14.19	11.93	15.54	12.34	7.44	4.91	7.71	5.72	20.20	-0.19	54
81.90	14.26	12.50	15.55	14.38	5.48	5.69	6.01	3.97	22.16	4.91	45
71.74	14.70	11.93	10.40	9.13	10.66	8.85	6.53	6.15	21.62	-2.16	87
73.61	13.37	11.87	13.22	9.56	10.16	10.21	4.09	4.20	23.31	-3.03	58
72.98	15.46	15.35	11.85	12.50	6.44	9.48	4.44	3.66	20.79	3.92	52
76.50	14.23	14.48	12.88	11.76	8.27	7.47	5.22	5.00	20.68	0.60	67.7

InceptionV3 came last as it got the worst overall performance, while the results of the second dataset in Table 10 showed that the VGG16 algorithm is the best-selected algorithm, and it is of close performance to the ResNet50 algorithm. InceptionV3 came last as it got the worst overall performance.

In the first dataset, the ResNet50 algorithm has the highest score of AUC 1317.770678, CAR 1404.603748, *F1*-score 1017.648389, precision 936.6420965, recall 824.2043559, FPR 711.7677382, PPV 537.504169, NPV 390.2033404, and final result 5715.987691. The final results represent the summation of all criteria results and subtracting the MSE. The second-best score is achieved by the

GoogleNet algorithm and the InceptionV3 algorithm has the lowest scores among the 15 algorithms. In the second dataset, the VGG16 algorithm has the highest score of AUC 1299.884286, CAR 1302.852674, *F1*-score 1143.826591, precision 1036.285298, recall 735.858182, FPR 664.348122, PPV 465.3038828, NPV 439.9139883, MSE 0.785912024 (lowest), and final result 5758.790868. The second-best score is achieved by the ResNet50 algorithm and the InceptionV3 algorithm also has the lowest scores among the 15 algorithms. Our analysis is an optimized combination of 10 tests and 15 deep learning methods for COVID-19 identification. The crow swarm optimization (CSO) is used to find the best set of coefficients that can be applied through a designed

TABLE 10: Applying CSO results to evaluate the deep learning algorithms (first dataset).

Algorithm	AUC	CAR	F1-score	Precision	Recall	FPR	PPV	NPV	MSE	Final result
<i>The results of the first dataset</i>										
ResNet50	1317.77	1404.60	1017.64	936.64	824.20	711.77	537.50	390.20	0.82	5715.99
DarkNet	1174.64	1307.39	957.15	869.42	742.66	643.42	498.72	354.63	1.45	5259.72
GoogleNet	1262.75	1387.56	1005.62	941.65	835.05	696.11	538.99	390.16	0.93	5664.74
MobileNetV2	1240.69	1357.00	986.16	869.84	805.39	643.03	507.89	376.23	0.80	5499.36
Xception	1090.73	1184.53	843.67	803.34	693.35	605.69	443.13	315.73	1.98	4766.83
VGG19	1165.93	1295.87	947.92	808.25	867.02	643.03	498.30	353.94	1.71	5292.50
VGG16	1181.02	1230.76	902.49	824.32	736.35	612.11	476.80	332.82	1.64	5070.82
InceptionV3	920.32	1013.29	728.85	693.84	601.52	493.12	387.91	275.79	1.16	4127.24
ResNet34	1314.28	1393.09	905.08	827.56	742.28	702.92	537.44	389.21	0.88	5405.14
CNNs	1269.72	1353.77	937.46	913.89	814.47	676.07	524.22	375.14	1.20	5511.41
DNN	1210.49	1310.92	914.30	872.34	793.81	657.28	501.28	357.62	1.33	5302.16
SAE	1166.94	1261.47	898.78	885.91	759.24	633.87	500.09	359.70	1.24	5197.02
InceptionResNetV2	1243.59	1350.70	968.39	914.83	780.65	674.35	525.41	369.43	2.06	5476.59
LSTM	1281.04	1390.47	993.69	931.63	824.48	680.38	536.85	379.38	2.02	5655.16
NASNet-Large	1151.99	1230.30	888.21	813.26	724.31	593.95	473.46	359.00	1.66	5044.91
<i>The results of the second dataset</i>										
ResNet50	1293.48	1290.40	1124.24	1041.93	725.85	664.05	463.27	451.61	1.12	5725.61
DarkNet	1170.09	1219.87	1075.52	967.16	681.18	628.43	434.35	412.17	1.55	5330.35
GoogleNet	1206.24	1256.51	1094.34	1001.61	704.01	640.08	434.98	421.77	1.22	5478.14
MobileNetV2	1182.90	1235.80	1085.06	991.85	690.69	633.73	444.22	416.07	0.97	5411.87
Xception	1207.23	1178.16	1032.60	940.36	665.37	603.64	419.26	404.87	1.26	5242.96
VGG19	1198.55	1230.58	1092.53	969.39	685.48	622.23	435.71	406.12	0.95	5395.18
VGG16	1299.88	1302.85	1143.83	1036.29	735.86	664.35	465.30	439.91	0.79	5758.79
InceptionV3	1027.64	1091.12	951.28	863.71	610.77	559.06	377.61	374.17	1.43	4735.81
ResNet34	1286.79	1277.51	1114.57	975.39	721.55	653.60	456.17	381.32	0.99	5558.71
CNNs	1213.79	1211.47	1071.01	970.45	676.38	615.29	432.05	413.67	1.03	5372.49
DNN	1166.39	1209.30	1061.60	977.15	679.36	620.67	427.72	414.17	1.72	5313.30
SAE	1182.62	1200.61	1055.16	945.53	672.32	611.11	424.69	406.87	1.61	5275.07
InceptionResNetV2	1257.76	1279.54	1126.81	1019.94	714.93	653.60	459.61	437.26	1.43	5640.84
LSTM	1173.23	1205.53	1061.08	970.80	687.38	618.05	430.90	396.17	2.05	5305.00
NASNet-Large	1154.30	1192.64	1037.24	942.94	667.44	607.15	426.36	410.77	1.28	5223.26

fitness function to evaluate the performance of the used deep learning algorithms. CSO is designed to ensure a good distribution for all the selected coefficients. This operation took the best available average fitness (not just the best one). The CSO algorithm is utilized to assess the diverse deep learning approaches for COVID-19 regarding the assessment measures. The research outcomes revealed that the selection problem related to COVID-19 identification methods could be viably solved utilizing the CSO approach.

This work concerns with early classification of COVID-19 as it is important for the treatment and control of diseases. Compared to reverse transcription-polymerase chain reaction (RT-PCR), chest computed tomography (CT) imaging can be a much more accurate, effective, and rapid technique for classifying and assessing COVID-19, especially in the pandemic area. Almost all hospitals have CT imaging machines; thus, CT images in the chest can be used for the early diagnosis of COVID-19 patients. The CT-based chest diagnosis of COVID-19 requires a specialist in radiology and a substantial amount of time, which is useful when the outbreak of COVID-19 is through at a rapid rate. Automated analysis of chest CT images is therefore desirable to save the precious time of medical professionals. In the meantime, for predicting cases of COVID-19, several deep learning models have been suggested. There is a need for a

choice solution for the optimized COVID-19 deep learning models, which was considered the key issue. In addition, there is no single study to address the issue of optimizing the COVID-19 diagnostic model based on pretrained and trained CNN and in-depth learning. Nevertheless, our research proposed an intellectual framework to assist medical centers and hospitals in selecting the COVID-19 diagnostic model. The assessment of identification approaches for COVID-19 is not a trivial task. Multiple measurements must be evaluated, and some of the measures conflicted with each other.

Our study is mainly constrained by the limitations of the available COVID-19 datasets. As explored in the related studies, the deep learning models need a high and large number of CT lung samples. The selection approach for optimal deep learning COVID-19 diagnostic model based on a novel CSO algorithm can be used for solving complex cases of different related studies to find optimal models. In future directions, the following aspects must be considered to enhance the outcomes of COVID-19 diagnostics: first, suggest to use more trained deep learning models that help evaluate and select the proposed approach for optimal deep learning COVID-19 diagnostic model based on a novel CSO algorithm. Second, using more measurements to support and evaluate the selection approach for optimal deep



learning COVID-19 diagnostic model based on a novel CSO algorithm. In addition, the main limitation of the CSO algorithm, like all swarm algorithms, is no guarantee to find the optimal solution. This is due to the nature of the problem search space.

## 5. Conclusion

Because of the COVID-19 pandemic, the number of computerized COVID-19 diagnosis studies is growing rapidly. This raises the question of which decision-makers should select the optimal COVID-19 diagnostic system in healthcare organizations and which performance criteria should be considered. Because of this, a selection scheme is necessary to address all the above issues. This study aims to bridge the gap between COVID-19 diagnostic model and deep learning models. To the best of the authors' knowledge, there is no work addressed and investigated the selection of the optimal COVID-19 diagnosis models. Only a review of main evaluation measurements for diagnostic COVID-19 deep learning-based models is presented in the scientific literature without addressing bridging the gap between the deep learning models and selection strategy for the optimal model. The crow swarm optimization (CSO) is employed to find an optimal set of coefficients using a designed fitness function for evaluating the performance of the deep learning model. The CSO is modified to obtain a well-selected coefficient distribution by considering the best average fitness. In this study, we present a description of the mechanism of the proposed methodology development. The outcomes that integrated a mix of 15 deep learning-based methods for COVID-19 diagnosis and nine evaluation metrics are presented in this article. Despite some challenges, including that no similar mechanism has been used to identify the significance of evaluation metrics ranking the deep learning-based models concerning different evaluation metrics is not an easy task, especially considering that some metrics conflict with each other. Our work will address all the challenges when developing the integrated framework. With a massive number of COVID-19 diagnostic deep learning-based models, it is not a trivial task for healthcare managers to decide which model meets their requirements with respect to reliability, cost, and speed. This is the main challenge of our study. The CSO algorithm is utilized to assess the diverse deep learning approaches for COVID-19 regarding the assessment measures. The research outcomes revealed that the selection problem related to COVID-19 identification methods could be viably solved utilizing the CSO optimization approach. For the first dataset, the ResNet50 algorithm is the optimal deep learning model is selected as the ideal identification approach for COVID-19 with the closeness overall fitness value of 5715.988 for COVID-19 CT lung images case considered differential advancement. In contrast, the VGG16 algorithm is the optimal deep learning model. It is selected as the ideal identification approach for COVID-19 with the closeness overall fitness value of 5758.791 for the second dataset. Furthermore, selecting an inappropriate COVID-19 diagnostic model might be non-cost effective for medical and health institutions with a

strong need for a rapid and accurate diagnostic model. Our proposed methodology will help healthcare managers assess and evaluate the COVID-19 diagnostic model and select the optimal model that fits their requirements by saving time, cost, and effort and obtain accurate and reliable results. Our proposed methodology can be used to evaluate a diagnostic model that uses the chest X-ray image to assist healthcare administrators in deciding which is the best COVID-19 diagnostic model.

## Data Availability

Data derived from public domain resources are in references [19, 38].

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Authors' Contributions

All authors contributed to the work and they read and approved the final manuscript.

## Acknowledgments

This research was supported by Basque Country Government.

## References

- [1] Z. A. A. Alyasseri, M. A. Al Betar, I. A. Doush, M. A. Awadallah, and A. K. Abasi, "Review on COVID 19 diagnosis models based on machine learning and deep learning approaches," *Expert Systems*, vol. 39, Article ID 12759, 2021.
- [2] J. N. Hasoon, A. H. Fadel, R. S. Hameed et al., "COVID-19 anomaly detection and classification method based on supervised machine learning of chest X-ray images," *Results in Physics*, vol. 31, Article ID 105045, 2021.
- [3] H. Alloui, M. A. Mohammed, N. Benameur et al., "A multi-agent deep reinforcement learning approach for enhancement of COVID-19 CT image segmentation," *Journal of Personalized Medicine*, vol. 12, no. 2, p. 309, 2022.
- [4] A. T. Sahlol, D. Yousri, A. A. Ewees, M. A. A. Al-qaness, R. Damasevicius, and M. A. Elaziz, "COVID-19 image classification using deep features and fractional-order marine predators algorithm," *Scientific Reports*, vol. 10, no. 1, pp. 1–15, Article ID 15364, 2020.
- [5] S. Albahli, A. Algsham, S. Aeraj et al., "COVID-19 public sentiment insights: a text mining approach to the gulf countries," *Computers, Materials & Continua*, vol. 67, no. 2, pp. 1613–1627, 2021.
- [6] D. Yousri, M. Abd Elaziz, L. Abualigah, D. Oliva, M. A. Al-Qaness, and A. A. Ewees, "COVID-19 X-ray images classification based on enhanced fractional-order cuckoo search optimizer using heavy-tailed distributions," *Applied Soft Computing*, vol. 101, Article ID 107052, 2021.
- [7] A. Bernheim, X. Mei, M. Huang et al., "Chest CT findings in coronavirus disease-19 (COVID-19): relationship to duration of infection," *Radiology*, vol. 295, no. 3, Article ID 200463, 2020.

- [8] M. A. Al-Qaness, A. I. Saba, A. H. Elsheikh et al., "Efficient artificial intelligence forecasting models for COVID-19 outbreak in Russia and Brazil," *Process Safety and Environmental Protection*, vol. 149, pp. 399–409, 2021.
- [9] M. A. Al-Qaness, H. Fan, A. A. Ewees, D. Yousri, and M. Abd Elaziz, "Improved ANFIS model for forecasting Wuhan City air quality and analysis COVID-19 lockdown impacts on air quality," *Environmental Research*, vol. 194, Article ID 110607, 2021.
- [10] K. H. Abdulkareem, "Realizing an Effective COVID-19 Diagnosis System Based on Machine Learning and IOT in Smart Hospital Environment," *IEEE Internet of things journal*, vol. 8, 2021.
- [11] M. Abd Elaziz, M. Aa Al-Qaness, E. O. Abo Zaid, S. Lu, R. Ali Ibrahim, and A. A. Ewees, "Automatic clustering method to segment COVID-19 CT images," *PLoS One*, vol. 16, no. 1, Article ID 0244416, 2021.
- [12] L. Li, *Artificial Intelligence Distinguishes COVID-19 from Community Acquired Pneumonia on Chest CT*, Radiological society of north america, Illinois, IL, USA, Article ID 200905, 2020.
- [13] M. A. Mohammed, "A comprehensive investigation of machine learning feature extraction and classification methods for automated diagnosis of covid-19 based on x-ray images," *XXX*, vol. 66, no. 3, 2020.
- [14] C. Butt, J. Gill, D. Chun, and B. Babu, "Deep learning system to screen coronavirus disease 2019 pneumonia," *XXX*, vol. 1, 2020.
- [15] O. I. Obaid, M. A. Mohammed, and S. Mostafa, "Long short-term memory approach for coronavirus disease," *Predicti*, vol. 12, pp. 11–21, 2020.
- [16] X. Xie, Z. Zhong, W. Zhao, C. Zheng, F. Wang, and J. J. R. Liu, *Chest CT for Typical 2019-nCoV Pneumonia: Relationship to Negative RT-PCR Testing*, Article ID 200343, 2020.
- [17] I. D. Apostolopoulos, T. A. J. P. Mpesiana, and E. Medicine, "Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks," *Physical and Engineering Sciences in Medicine*, vol. 43, no. 2, pp. 635–640, 2020.
- [18] E. E.-D. Hemdan, M. A. Shouman, and M. Karar, "Covidx-net: A Framework of Deep Learning Classifiers to Diagnose Covid-19 in X-ray Images," 2020, <https://org/arXiv:2003.11055>.
- [19] J. P. Cohen, P. Morrison, L. Dao, K. Roth, T. Q. Duong, and M. Ghassemi, "COVID-19 Image Data Collection: Prospective Predictions Are the Future," 2020, <https://org/arXiv:2006.11988>.
- [20] A. Rosebrock, *Detecting COVID-19 in X-ray Images with Keras, TensorFlow, and Deep Learning*, California, USA, 2020.
- [21] M. Barstugan, U. Ozkaya, and S. Ozturk, "Coronavirus (Covid-19) Classification Using Ct Images by Machine Learning Methods," 2020, <https://arxiv.org/abs/2003.09424>.
- [22] L. Wang, A. Wong, and Z. Q. Lin, "COVID-net: A Tailored Deep Convolutional Neural Network Design for Detection of COVID-19 Cases from Chest X-Ray Images," *Scientific Reports*, vol. 10, Article ID 19549, 2020.
- [23] P. J. O. Mooney, "Chest x-ray images (pneumonia)," *XXX*, 2018.
- [24] H. S. Maghdid, A. T. Asaad, K. Z. Ghafoor, A. S. Sadiq, and M. K. J. a. p. a. Khan, "Diagnosing COVID-19 Pneumonia from X-ray and CT Images Using Deep Learning and Transfer Learning Algorithms," 2020, <https://arxiv.org/abs/2004.00038>.
- [25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.
- [26] B. Ghoshal and A. Tucker, "Estimating Uncertainty and Interpretability in Deep Learning for Coronavirus (COVID-19) Detection," 2020, <https://arxiv.org/abs/2003.10769>.
- [27] L. O. Hall, R. Paul, D. B. Goldgof, and G. Goldgof, "Finding Covid-19 from Chest X-Rays Using Deep Learning on a Small Dataset," 2020, <https://arxiv.org/ftp/arxiv/papers/2004/2004.02060.pdf>.
- [28] M. Farooq and A. Hafeez, "Covid-resnet: A Deep Learning Framework for Screening of COVID-19 from Radiographs," 2020, <https://arxiv.org/abs/2003.14395>.
- [29] Ş. Öztürk, U. Özkaya, M. Barstugan, and Technology, "Classification of Coronavirus (COVID-19) from X-ray and CT images using shrunken features," *International Journal of Imaging Systems and Technology*, vol. 31, no. 1, pp. 5–15, 2021.
- [30] A. Abbasian Ardakani, U. R. Acharya, S. Habibollahi, and A. Mohammadi, "COVIDdiag: a clinical CAD system to diagnose COVID-19 pneumonia based on CT findings," *European Radiology*, vol. 31, no. 1, pp. 121–130, 2021.
- [31] T. Javaheri, M. Homayounfar, Z. Amoozgar et al., "Covidctnet: An Open-Source Deep Learning Approach to Identify Covid-19 Using Ct Image," 2020, <https://arxiv.org/abs/2005.03059>.
- [32] V. Shah, R. Keniya, A. Shridharani, M. Punjabi, J. Shah, and N. Mehendale, "Diagnosis of COVID-19 using CT scan images and deep learning techniques," *Emergency Radiology*, vol. 28, no. 3, pp. 497–505, 2021/02/01 2021.
- [33] S. Wang, B. Kang, J. Ma et al., "A Deep Learning Algorithm Using CT Images to Screen for Corona Virus Disease (COVID-19)," *European Radiology*, vol. 31, pp. 6096–6104, 2021.
- [34] F. Zheng, L. Li, X. Zhang et al., "Accurately discriminating COVID-19 from viral and bacterial pneumonia according to CT images via deep learning," *Interdisciplinary Sciences: Computational Life Sciences*, vol. 13, pp. 273–285, 2021.
- [35] O. Gozes, M. Frid-Adar, H. Greenspan et al., "Rapid Ai Development Cycle for the Coronavirus (Covid-19) Pandemic: Initial Results for Automated Detection & Patient Monitoring Using Deep Learning Ct Image Analysis," 2020, <https://arxiv.org/abs/2003.05037>.
- [36] N. A. Baghdadi, A. Malki, S. F. Abdelaliem, H. Magdy Balaha, M. Badawy, and M. Elhosseini, "An automated diagnosis and classification of COVID-19 from chest CT images using a transfer learning-based convolutional neural network," *Computers in Biology and Medicine*, vol. 144, Article ID 105383, 2022.
- [37] S. Nabavi, A. Ejmalian, M. E. Moghaddam et al., "Medical imaging and computational image analysis in COVID-19 diagnosis: a review," *Computers in Biology and Medicine*, vol. 135, Article ID 104605, 2021.
- [38] Q. Li, J. Ning, J. Yuan, and L. Xiao, "A depthwise separable dense convolutional network with convolution block attention module for COVID-19 diagnosis on CT scans," *Computers in Biology and Medicine*, vol. 137, Article ID 104837, 2021.
- [39] A. d. F. Cobre, D. P. Stremel, G. R. Noleto et al., "Diagnosis and prediction of COVID-19 severity: can biochemical tests and machine learning be used as prognostic indicators?" *Computers in Biology and Medicine*, vol. 134, Article ID 104531, 2021.

- [40] F. Shan, Y. Gao, J. Wang et al., "Lung Infection Quantification of Covid-19 in Ct Images with Deep Learning," 2020, <https://arxiv.org/abs/2003.04655>.
- [41] F. B. Hamzah, C. Lau, H. Nazri, D. Ligot, G. Lee, and C. J. B. W. H. O. Tan, "CoronaTracker: worldwide COVID-19 outbreak data analysis and prediction," *XXX*, vol. 1, pp. 1–32, 2020.
- [42] S. A. Harmon, T. H. Sanford, S. Xu et al., "Artificial intelligence for the detection of COVID-19 pneumonia on chest CT using multinational datasets," *Nature Communications*, vol. 11, no. 1, pp. 4080–4087, 2020.
- [43] N. M. Kumar, M. A. Mohammed, K. H. Abdulkareem et al., "Artificial intelligence-based solution for sorting COVID related medical waste streams and supporting data-driven decisions for smart circular economy practice," *Process Safety and Environmental Protection*, vol. 152, pp. 482–494, 2021.
- [44] M. A. Mohammed, M. S. Maashi, M. Arif, M. K. Nallapaneni, and O. Geman, "Intelligent systems and computational methods in medical and healthcare solutions with their challenges during COVID-19 pandemic," *Journal of Intelligent Systems*, vol. 30, no. 1, pp. 976–979, 2021.
- [45] K. H. Abdulkareem, S. A. Mostafa, Z. N. Al-Qudsy et al., "Automated system for identifying COVID-19 infections in computed tomography images using deep learning models," *Journal of Healthcare Engineering*, vol. 2022, Article ID 5329014, 2022.
- [46] A. Lakhan, M. A. Mohammed, A. N. Rashid et al., "Smart-contract aware ethereum and client-fog-cloud healthcare system," *Sensors*, vol. 21, no. 12, p. 4093, 2021.
- [47] M. S. P. Subathra, M. S. P. Subathra, M. A. Mohammed et al., "Detection of focal and non-focal electroencephalogram signals using fast Walsh-Hadamard transform and artificial neural network," *Sensors*, vol. 20, no. 17, p. 4952, 2020.
- [48] S.-C. B. Lo, H.-P. Chan, J.-S. Lin, H. Li, M. T. Freedman, and S. K. J. N. n. Mun, "Artificial convolution neural network for medical image pattern recognition," *Neural Networks*, vol. 8, no. 7-8, pp. 1201–1214, 1995.
- [49] R.-C. J. I. Chen and R. C. Chen, "Automatic License Plate Recognition via sliding-window darknet-YOLO deep learning," *Image and Vision Computing*, vol. 87, pp. 47–56, 2019.
- [50] S. Han, X. Liu, H. Mao et al., "EIE: efficient inference engine on compressed deep neural network," *ACM SIGARCH - Computer Architecture News*, vol. 44, no. 3, pp. 243–254, 2016.
- [51] P. Ballester and R. M. Araujo, "On the performance of GoogLeNet and AlexNet applied to sketches," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pp. 1124–1128, Phoenix, Arizona, February 2016.
- [52] C. A. Ferreira, T. Melo, P. Sousa et al., "Classification of breast cancer histology images through transfer learning using a pre-trained inception resnet v2," *International Conference Image Analysis and Recognition*, vol. 10882, pp. 763–770, 2018.
- [53] U. P. Singh, S. S. Chouhan, S. Jain, and S. J. I. A. Jain, "Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease," *IEEE Access*, vol. 7, Article ID 43721, 2019.
- [54] P. Zhou, J. Han, G. Cheng, B. Zhang, and R. Sensing, "Learning compact and discriminative stacked autoencoder for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 7, pp. 4823–4833, 2019.
- [55] S. Rani and P. Kumar, "Deep learning based sentiment analysis using convolution neural network," *Arabian Journal for Science and Engineering*, vol. 44, no. 4, pp. 3305–3314, 2019.
- [56] E. A. Smirnov, D. M. Timoshenko, and S. N. J. A. P. Andrianov, "Comparison of regularization methods for imagenet classification with deep convolutional neural networks," *AASRI Procedia*, vol. 6, pp. 89–94, 2014.
- [57] Y. Chen, H. Jiang, C. Li, X. Jia, P. Ghamisi, and R. Sensing, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 6232–6251, 2016.
- [58] A. Lakhan, Q.-U.-A. Mastoi, M. Elhoseny, M. S. Memon, and M. Mohammed, "Deep neural network-based application partitioning and scheduling for hospitals and medical enterprises using IoT assisted mobile fog cloud," *Enterprise Information Systems*, vol. 16, pp. 1–23, 2021.
- [59] M. Yousif and B. Al-Khateeb, "A novel metaheuristic algorithm for multiple traveling salesman problem," *Jardics*, vol. 10, no. 13, pp. 2113–2122, 2018.
- [60] A. Nickabadi, M. M. Ebadzadeh, and R. Safabakhsh, "A novel particle swarm optimization algorithm with adaptive inertia weight," *Applied Soft Computing*, vol. 11, no. 4, pp. 3658–3670, 2011.
- [61] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.
- [62] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: algorithm and applications," *Future Generation Computer Systems*, vol. 97, pp. 849–872, 2019.
- [63] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. Mirjalili, "Salp Swarm Algorithm: a bio-inspired optimizer for engineering design problems," *Advances in Engineering Software*, vol. 114, pp. 163–191, 2017.
- [64] S. Mirjalili and A. Lewis, "The Whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.

## Research Article

# DCNN-FuzzyWOA: Artificial Intelligence Solution for Automatic Detection of COVID-19 Using X-Ray Images

Abbas Saffari <sup>1</sup>, Mohammad Khishe <sup>1</sup>, Mokhtar Mohammadi <sup>2</sup>,  
Adil Hussein Mohammed <sup>3</sup> and Shima Rashidi <sup>4</sup>

<sup>1</sup>Department of Electrical Engineering, Imam Khomeini Marine Science University, Nowshahr, Iran

<sup>2</sup>Department of Information Technology, College of Engineering and Computer Science, Lebanese French University, Erbil, Kurdistan Region, Iraq

<sup>3</sup>Department of Communication and Computer Engineering, Faculty of Engineering, Cihan University-Erbil, Erbil, Kurdistan Region, Iraq

<sup>4</sup>Department of Computer Science, College of Science and Technology, University of Human Development, Sulaymaniyah, Kurdistan Region, Iraq

Correspondence should be addressed to Abbas Saffari; [abbas.saffari@birjand.ac.ir](mailto:abbas.saffari@birjand.ac.ir)

Received 9 February 2022; Revised 1 June 2022; Accepted 14 June 2022; Published 9 August 2022

Academic Editor: Ahmed A. Ewees

Copyright © 2022 Abbas Saffari et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Artificial intelligence (AI) techniques have been considered effective technologies in diagnosing and breaking the transmission chain of COVID-19 disease. Recent research uses the deep convolution neural network (DCNN) as the discoverer or classifier of COVID-19 X-ray images. The most challenging part of neural networks is the subject of their training. Descent-based (GDB) algorithms have long been used to train fully connected layer (FCL) at DCNN. Despite the ability of GDBs to run and converge quickly in some applications, their disadvantage is the manual adjustment of many parameters. Therefore, it is not easy to parallelize them with graphics processing units (GPUs). Therefore, in this paper, the whale optimization algorithm (WOA) evolved by a fuzzy system called FuzzyWOA is proposed for DCNN training. With accurate and appropriate tuning of WOA's control parameters, the fuzzy system defines the boundary between the exploration and extraction phases in the search space. It causes the development and upgrade of WOA. To evaluate the performance and capability of the proposed DCNN-FuzzyWOA model, a publicly available database called COVID-Xray-5k is used. DCNN-PSO, DCNN-GA, and LeNet-5 benchmark models are used for fair comparisons. Comparative parameters include accuracy, processing time, standard deviation (STD), curves of ROC and precision-recall, and F1-Score. The results showed that the FuzzyWOA training algorithm with 20 epochs was able to achieve 100% accuracy, at a processing time of 880.44 s with an F1-Score equal to 100%. Structurally, the i-6c-2s-12c-2s model achieved better results than the i-8c-2s-16c-2s model. However, the results of using FuzzyWOA for both models have been very encouraging compared to particle swarm optimization, genetic algorithm, and LeNet-5 methods.

## 1. Introduction

COVID-19 was initially designated an epidemic disease by the World Health Organization (WHO) in March 2020 [1]. Due to the increasing number of deaths, the spread of the disease, the lack of access to vaccines and particular drugs, and rapid diagnosis of the disease to break, the transmission chain has become one of the most important research topics for researchers. Polymerase chain reaction (PCR) test [2] and X-ray images [3] are standard methods in detecting

COVID-19. One of the problems of PCR tests is that there are not enough kits and also it takes a relatively long time to answer the test. In addition to being affordable, X-ray images are always and everywhere available. Reducing the time to diagnose and detect positive cases, even without fever and cough symptoms, are other benefits of using X-ray images [4]. AI tools can increase processing time and high accuracy in detecting patients with COVID-19 [5]. Much research has been done to identify positive cases of COVID-19 [3, 6]. However, until COVID-19 disease is completely eradicated,

the need to research and discover new, fast, low-cost, and accurate techniques is acute. DL is one of the AI techniques for detecting positive cases of COVID-19 [7]. Training is the most challenging part of DL. Examples of algorithms used for DL training are conjugate gradient (CG) algorithm [8], Krylov subspace descent (KSD) algorithm [9], and Hessian-free optimization (HFO) approach [10].

While stochastic GDB training methods are simple to construct and run quickly in the producer for large numbers of training samples, GDB approaches require extensive manual parameter adjustment for optimal performance. Their structure is sequential and leads to parallelizing them with GPU become challenging. On the other hand, though CG methods are stable for training, they are almost slow lead to needing multiple CPUs and a lot of RAMs resource [8]. Deep auto-encoders used HFO to train the weights of standard CNNs, which performs better than Hinton and Salakhutdinov's approach for pretraining and fine-tuning deep auto-encoders [11]. In addition, HFO is weaker than KSD and more complex. In terms of the amount of memory required, HFO requires less memory than KSD. KSD optimization and classification speeds also work better [9]. Recent years have seen the employment of metaheuristic and evolutionary algorithms to solve and optimize real-world problems [12–14]. Despite this, research on optimizing DL training needs to be given more attention. Optimization based on metaheuristic algorithms with a hybrid genetic algorithm and DCNN is the beginning of this field study [15]. This model determines the DCNN parameters through GA's crossover and mutation processes, with the DCNN structure modeled as a chromosome in GA. Alternatively, only the weights and biases of the first convolution layer (C1) and the third convolution layer (C3) are used as chromosomes during the crossover step. In [16], they present an evolutionary method for fine-tuning the parameters of a DCNN by utilizing the Harmony Search (HS) algorithm and several of its improved variants for handwritten field digit and fingerprint detection. In [17], researchers will develop a hybrid deep neural network (DNN), using computed tomography (CT) and X-ray imaging, to predict the risk of COVID-19-related disease onset. In [18], a new method of diagnosing COVID-19 based on chest X-ray images using artificial intelligence is proposed. In comparison to the state-of-the-art techniques currently used, the proposed method will demonstrate outstanding performance.

In [19], the progressive unsupervised learning (PAUL) algorithm is used for DCNN training. PUL is the easiest way to implement. Therefore, it is considered a primary benchmark for unsupervised feature learning. Due to the fact that clustering data sets might be difficult to categorize, PUL initially inserts a selection stage between the clustering and fine-tuning stages. In [20], an approach for automatically building DCNN architectures on the basis of GA is suggested for optimizing image classification. The lack of knowledge about the structure of DCNN is the most crucial feature of this method. In contrast, the presence of large DCNNs causes chromosomes to grow, thus slowing down the algorithm. Due to the faults described, our proposed strategy comprises training a DCNN model on Data 1 to

identify positive and negative cases of COVID-19 samples using X-ray pictures. Following that, the previously trained DCNN's FCL will be replaced with the new FCL, which has been tuning using the whale optimization algorithm, and employs fuzzy logic to adjust its control parameters for better WOA development and performance. The name of the proposed algorithm is called FuzzyWOA. Therefore, in this article, our main motivation is to investigate the impact of FuzzyWOA on improving DCNN performance. Our main contribution in this paper is to improve WOA performance by designing and applying a fuzzy system to balance the exploration and extraction boundaries in the search space for automatic detection of COVID-19 using X-ray images. In this regard, for a fairer comparison, in addition to FuzzyWOA, PSO, GA, and LeNet-5 are used for two DCNN models with different structures in order to automatically detect COVID-19 cases. Of course, it should be noted that various metaheuristic methods have been used to train the neural network, such as sine-cosine algorithm [21], Salp swarm algorithm [22], best-mass gravitational search algorithm [23], particle swarm optimizer [24], biogeography-based optimization [25], dragonfly algorithm [26], and chimp optimization algorithm [27]. But the common problem of these algorithms that leads to inefficiency in some problems is the lack of detection of two phases of exploration and extraction. One of the advantages of using FuzzyWOA is establishing a correct trade-off between the two phases of exploration and extraction in the algorithm's search space. Other disadvantages of using some high metaheuristic methods include being stuck in local optimizations, low convergence speed, high complexity, increasing the number of control parameters, and so on. For this reason, it seems necessary to use an algorithm that performs better in less time. Improvements to FuzzyWOA have eliminated all of these drawbacks. Following that, the other connection weights are kept in the residual layers of the pretrained DCNN, resulting in the training of a linear structure using the characteristics of the final layer.

## 2. Materials and Methods

This section consists of four subsections. The first subsection first introduces WOA and then describes the proposed FuzzyWOA algorithm. The second subsection deals with the DCCN model. The third subsection is about the COVID X-ray database, and the fourth subsection describes the methodology.

*2.1. FuzzyWOA.* First, the WOA mathematical model is explained, and then how to use fuzzy logic to develop the algorithm.

*2.1.1. WOA.* The WOA optimization algorithm was introduced in 2016, inspired by the way whales were hunted by Mirjalili and Lewis [28]. WOA begins with a collection of randomly generated solutions. Each iteration, the search agents update their location by using three operators: encircling prey, bubble-net assault (extraction phase), and

bait search (exploration phase). Whales discover and encircle prey. The WOA assumes that the best solution right now his prey. That once best search agent has been recognized, all other search agents' locations will be updated to point to the best search agent. This behavior is expressed by the following equations:

$$\vec{D} = |\vec{C} \cdot \vec{X}^*(t) - \vec{X}(t)|, \quad (1)$$

$$\vec{X}(t+1) = \vec{X}^*(t) - \vec{A} \cdot \vec{D}, \quad (2)$$

where  $t$  is the current iteration,  $\vec{A}$  and  $\vec{C}$  are the coefficient vectors,  $(\vec{X}^*)$  is the place vector is the best solution so far, and  $\vec{X}$  is the place vector. In each iteration of the algorithm,  $(\vec{X}^*)$  should be updated if a better answer is reached. The vectors  $\vec{A}$  and  $\vec{C}$  are obtained using the following equations:

$$\vec{A} = 2\vec{\alpha} \cdot \vec{r} - \vec{\alpha}, \quad (3)$$

$$\vec{C} = 2 \cdot \vec{r}, \quad (4)$$

where  $\vec{\alpha}$  decreases linearly from 2 to zero during repetitions and  $\vec{r}$  is a random vector in the distance  $[0, 1]$ . The whale uses the bubble-net assault strategy to swim simultaneously around its target and along a contraction circle in a spiral pattern. To describe this concurrent behavior, it is anticipated that the whale would change its location during optimization via one of the contractile siege mechanisms or the spiral model with a 50% probability. Equation (5) defines the mathematical model for this phase.

$$\vec{X}(t+1) = \begin{cases} \vec{X}^*(t) - \vec{A} \cdot \vec{D} & \text{if } p < 0.5 \\ 2\vec{D} \cdot e^{bi} \cdot \cos(2\pi l) & \text{if } p \geq 0.5, \end{cases} \quad (5)$$

where  $\vec{D}$  is obtained from equation (6) and refers to the distance  $i$  from the whale to the prey (the best solution ever obtained). A constant  $b$  is used to specify the geometry of the logarithmic helix, and  $l$  is a random value between  $-1$  and  $1$ .  $p$  is a nonzero integer between  $0$  and  $1$ . Vector  $A$  is used with random values between  $-1$  and  $1$  to bring search agents closer to the reference whale. In the search for prey to update the search agent's position, random agent selection is used instead of using the best search agent's data. The mathematical model is in the form of the following equations:

$$\vec{D} = |\vec{C} \cdot \vec{X}_{rand} - \vec{X}|, \quad (6)$$

$$\vec{X}(t+1) = \vec{X}_{rand} - \vec{A} \cdot \vec{D}, \quad (7)$$

$\vec{X}_{rand}$  is the randomly chosen position vector (random whale) for the current population, and vector  $A$  is utilized with random values larger or equal to one to drive the search agent away from the reference whale [29].

**2.1.2. Proposed Fuzzy Logic for Tuning Control Parameters.** The proposed fuzzy model receives the normalized performance of each whale in the population (normalized

fitness value) and the current values of the parameters  $\vec{\alpha}$  and  $\vec{C}$ . The output also shows the amount of change using the symbols  $\Delta\alpha$  and  $\Delta C$ . The NFV value for each whale is obtained by equation (8).

$$NFV = \frac{\text{fitness} - \text{fitness}_{\min}}{\text{fitness}_{\min} - \text{fitness}_{\max}}. \quad (8)$$

The NFV value is in the range of  $[0,1]$ . This paper's optimization problem is of the minimization type, in which the fitness of each whale is obtained directly by the optimal amount of these functions. Equations (9) updating the parameters  $\vec{\alpha}$  and  $\vec{C}$  for each whale are as follows:

$$\begin{aligned} \vec{\alpha}^{t+1} &= \vec{\alpha}^t + \Delta\alpha \\ \vec{C}^{t+1} &= \vec{C}^t + \Delta C. \end{aligned} \quad (9)$$

The fuzzy system is responsible for updating the parameters  $\vec{\alpha}$  and  $\vec{C}$  of each member of the population (whale), and the three inputs of this system are the current value of parameters  $\vec{\alpha}$ ,  $\vec{C}$ , and NFV. Initially, these values are "fuzzification" by membership functions. Then their membership value is obtained using  $\mu$ . These values are applicable to a set of rules and result in the values  $\Delta\alpha$  and  $\Delta C$ . Following the determination of these values, the "defuzzification" technique is used to approximate the numerical values  $\Delta\alpha$  and  $\Delta C$ . Finally, these values are applied in equations (9) and (10) to update the parameters  $\Delta\alpha$  and  $\Delta C$ . The fuzzy system used in this article is of the Mamdani type (see Table 1). The suggested fuzzy model and membership functions used to update the whale algorithm's control parameters are shown in Figure 1.

**2.2. Convolutional Neural Network.** DCNNs are very similar to multilayer perceptron neural networks [30]. These networks are built on the basis of three principles: weight sharing between connections, local receive fields, and temporal/spatial subsampling [31, 32]. The principles discussed above may be classified into two types of layers: subsampling layers and convolution layers. Three convolution layers  $C1$ ,  $C3$ , and  $C5$ , positioned between layers  $S2$  and  $S4$ , and a final output layer  $F6$  comprise the processing layers (as shown in Figure 2). Feature maps are used to arrange these subsampling and convolution layers. In the last layer, neurons in the convolution layer are connected to a local receptive field. Thus, neurons with the same feature maps (FMs) receive data from different input regions until the input is wholly skimmed to share identical weights. The FMs are spatially downsampled by a factor of two in the subsampling layer. For example, in subsequent layer  $S4$ , FM of size  $10 \times 10$  is subsampled to conforming FM of size  $5 \times 5$ . The last layer is responsible for categorization ( $F6$ ). Each FM in this structure is the result of convolution between the maps of the previous layer and their respective kernel and a linear filter. The weights  $w^k$  and adding bias  $b_k$  produce the  $k^{\text{th}}$  (FM)  $FM_i^k$  using the tanh function as equation (10).

TABLE 1: Applied fuzzy rules.

If (NFV is low) and ( $\vec{\alpha}$ is low), then ( $\Delta\alpha$ is ZE)
If (NFV is low) and ( $\vec{\alpha}$ is medium), then ( $\Delta\alpha$ is NE)
If (NFV is low) and ( $\vec{\alpha}$ is high), then ( $\Delta\alpha$ is NE)
If (NFV is medium) and ( $\vec{\alpha}$ is low), then ( $\Delta\alpha$ is PO)
If (NFV is medium) and ( $\vec{\alpha}$ is medium), then ( $\Delta\alpha$ is ZE)
If (NFV is medium) and ( $\vec{\alpha}$ is high), then ( $\Delta\alpha$ is NE)
If (NFV is high) and ( $\vec{\alpha}$ is low), then ( $\Delta\alpha$ is PO)
If (NFV is high) and ( $\vec{\alpha}$ is medium), then ( $\Delta\alpha$ is ZE)
If (NFV is high) and ( $\vec{\alpha}$ is high), then ( $\Delta\alpha$ is NE)
If (NFV is low) and ( $\vec{C}$ is low), then ( $\Delta C$ is PO)
If (NFV is low) and ( $\vec{C}$ is medium), then ( $\Delta C$ is PO)
If (NFV is low) and ( $\vec{C}$ is high), then ( $\Delta C$ is ZE)
If (NFV is medium) and ( $\vec{C}$ is low), then ( $\Delta C$ is PO)
If (NFV is medium) and ( $\vec{C}$ is medium), then ( $\Delta C$ is ZE)
If (NFV is medium) and ( $\vec{C}$ is high), then ( $\Delta C$ is NE)
If (NFV is high) and ( $\vec{C}$ is low), then ( $\Delta C$ is PO)
If (NFV is high) and ( $\vec{C}$ is medium), then ( $\Delta C$ is ZE)
If (NFV is high) and ( $\vec{C}$ is high), then ( $\Delta C$ is NE)

$$FM_{ij}^k = \tanh\left(\left(W^k \times x\right)_{ij} + b_k\right). \quad (10)$$

By lowering the resolution of FMs, the subsampling layer achieves spatial invariance, in which each pooled FM corresponds to a single FM in the previous layer. Equation (11) is defined as the subsampling function.

$$\alpha_j = \tanh\left(\beta \sum_{N \times N} \alpha_i^{n \times n} + b\right). \quad (11)$$

where  $\alpha_i^{n \times n}$  denotes the inputs and  $\beta$  and  $b$ , respectively, denote the trainable scalar and bias. After many convolution and subsampling layers, the final layer is a completely linked structure that carries out the classification process. Each output class has its own neuron. As a result, in the COVID-19 data set, this layer comprises two neurons for each of its classes.

**2.3. Data set.** The database used with the name COVID-X-ray-5k consists of 2084 tutorials and 3100 test images [33]. In this data set, since lateral images are not suitable for identifying the target and according to the radiologist's recommendations, anterior-posterior COVID-19 X-ray images have been used. Radiologists evaluate data set images, and items that do not have exact COVID-19 symptoms are removed. Out of 203 images, 19 images will be deleted, and 184 images with clear signs of COVID-19 will remain. By doing the job in this manner, the community was introduced, as well as a more clearly labeled data set. Of the remaining images, 184 images were used, 100 images were used for network testing, and 84 images were used for network training. Using data augmentation, we increase the number of COVID-19 samples to 420 samples. Due to the small amount of non-COVID pictures in the COVID-chest ray-data set [34], the supplemental ChexPert data set [35] was used. This data set contains 224316 chest X-ray images from 65240 individuals. Totally, 2000 images from the non-COVID-19 data set are used for the training set, and 3000 images are used for the test set. Table 2

summarizes the total number of photos utilized across all classes (see Table 2 and Figure 3).

Figure 3 illustrates two picture samples from COVID-19 and four standard image samples randomly picked from the COVID-X-ray-5k data set.

## 2.4. Methodology

**2.4.1. Presentation of Whales.** Two fundamental concepts govern the tuning of deep artificial neural networks: to begin, the structure's parameters must be accurately represented by a FuzzyWOA (candid solution); next, the fitness function must be defined in terms of the problem at hand. The use of FuzzyWOA in DCNN tuning is a distinct phase in the presentation of network parameters. Therefore, to achieve the highest and highest detection accuracy, the essential parameters in DCNN, i.e., weights and FCL, must be clearly defined. In general, FuzzyWOA optimizes the weights and biases used to compute the loss function as the fitness function in the final layer. In other words, whales are used in FuzzyWOA as the last layer's weight and bias values. Three main ways are available for representing the weights and biases of a DCNN as frank solutions of a metaheuristic algorithm: based on vectors, matrices, or binary states [26]. Since FuzzyWOA requires a vector-based model's parameters, this paper uses equation (12) for the candidate solution.

$$\text{Whales} = [W_{11} \cdot W_{12} \cdot \dots \cdot W_{nh} \cdot b_1 \cdot \dots \cdot b_h \cdot M_{11} \cdot \dots \cdot M_{hm}], \quad (12)$$

where  $n$  denotes the number of input nodes,  $W_{ij}$  denotes the weight of the connection between the  $i_{th}$  input node and the  $j_{th}$  hidden neuron,  $b_j$  denotes the bias of the  $j_{th}$  hidden neuron, and  $M_{jo}$  denotes the weight of the connection between the  $j_{th}$  hidden neuron and the  $o_{th}$  output neuron. As indicated in Section 2.2, the suggested design is a straightforward LeNet-5 framework. Two structures are utilized in this section: i-6c-2s-12c-2s and i-8c-2s-16c-2s, where C and S denote convolution and subsampling layers, respectively. All convolution layers have a kernel size of  $5 \times 5$ , and the scale of subsampling is downsampled by a factor of two.

**2.5. Loss Function.** In designing and proposing the proposed metaheuristic optimizer (DCNN-FuzzyWOA), the task of DCNN training is the responsibility of FuzzyWOA. The purpose of optimization is to obtain the best accuracy, minimizing classification error and network complexity. This target may be calculated using either the whales' loss function or the classification procedure's mean square error (MSE). As a result, the lost function is defined as equation (13).

$$y = \frac{1}{2} \sqrt{\frac{\sum_{i=0}^N (o - d)^2}{N}}, \quad (13)$$

where  $o$  denotes the computed output,  $d$  is the desired output, and  $N$  denotes the training sample count. Two

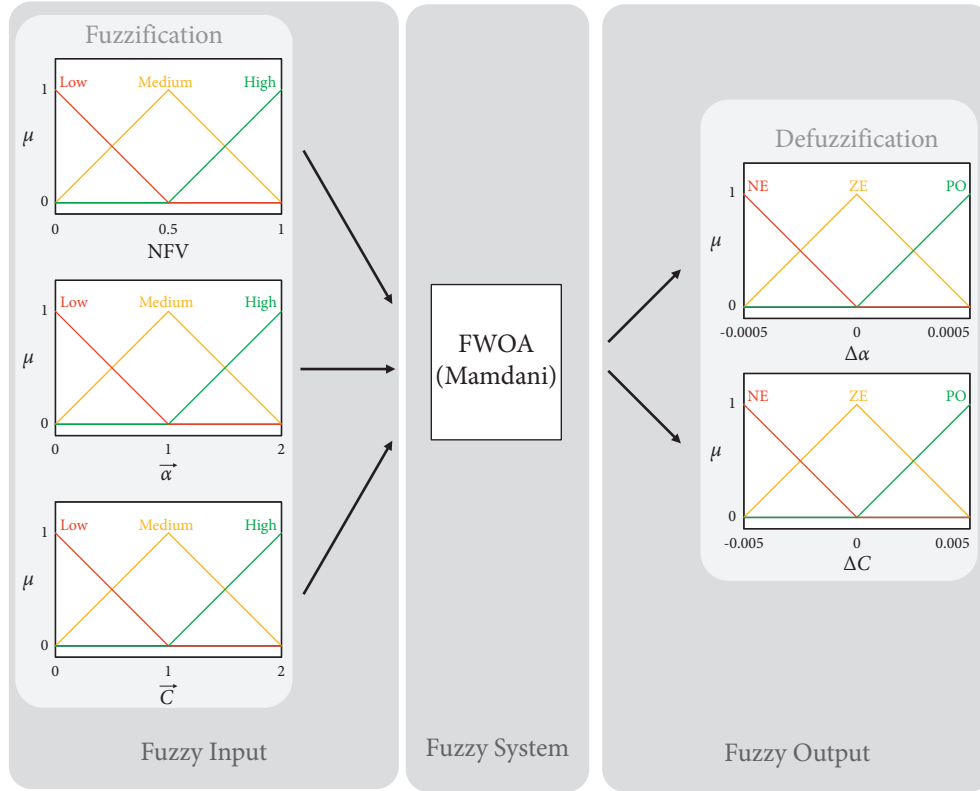
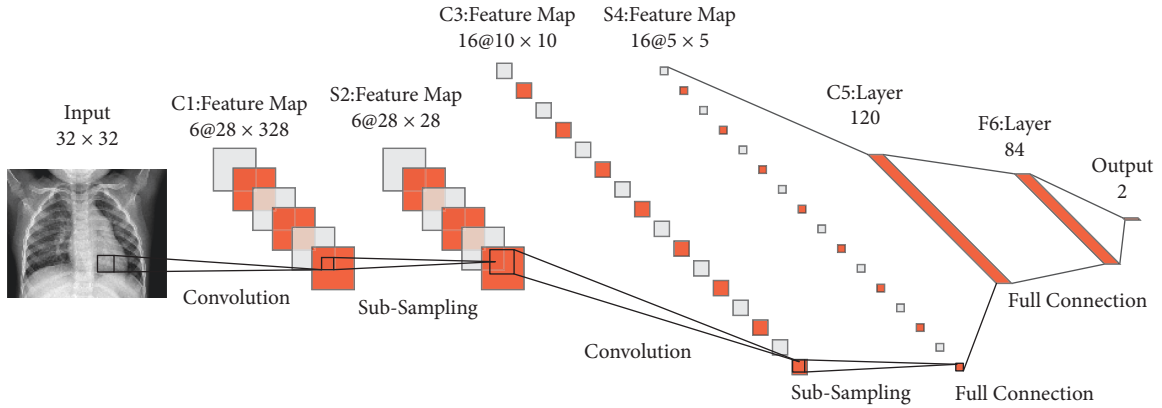

 FIGURE 1: A proposed fuzzy model for setting parameters  $\vec{\alpha}$  and  $\vec{C}$ .


FIGURE 2: The LeNet-5 DCNN's architecture.

TABLE 2: The COVID data set's image categories [33].

Category	COVID-19	Normal
Training set	84 (420 after augmentation)	2000
Test set	100	3000

conditions are defined to terminate FuzzyWOA, including reaching maximum iteration or predefined loss function.

### 3. Results and Discussion

As mentioned in the previous sections, this paper attempts to improve the classic DCNN-FuzzyWOA classifier's

accuracy by proposing and designing a fuzzy system to adjust the WOA control parameters. For the DCNN-FuzzyWOA simulation, the population size and maximum iteration are 15. In DCNN, the batch size is 100, and the learning rate is 1. Additionally, the number of epochs examined for each assessment ranges between 1 and 20. The test was conducted in MATLAB-R2020a on a PC equipped with an Intel Core i7-2630QM CPU and 6 GB of RAM running Windows 7, with six distinct runtimes. According to reference [20], the accuracy rate cannot provide sufficient information about the detector's effectiveness.

The suggested classifier's effectiveness in all samples was shown using receiver operating characteristic (ROC) curves.



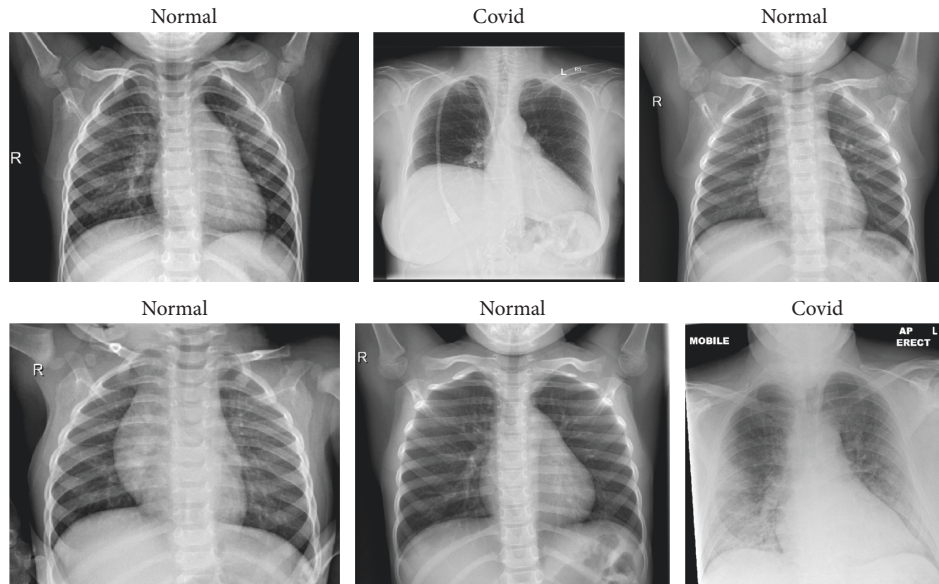


FIGURE 3: Images random from the COVID-X-ray-5k data set [33].

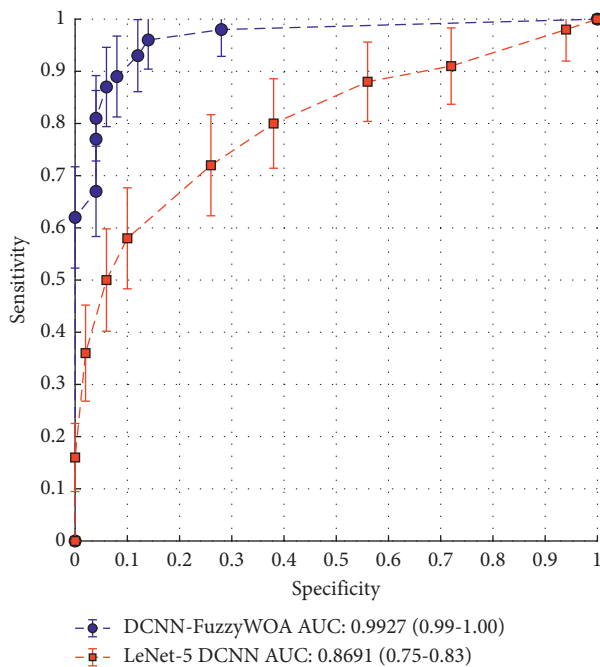


FIGURE 4: ROC curves for DCNN-FuzzyWOA and LeNet-5.

As a result, each sample is assigned an estimated probability of images  $P_T$ . Following that, a threshold value  $T \in [0.1]$  was added. Thus, the detection rate was determined for each value. Thus, the obtained values were presented as a receiver operating characteristic (ROC) curve. In general, the concept of ROC diagram curves can be interpreted so that the larger the area under the diagram (AUC), the greater the probability of detection. Figure 4 shows the result of the ROC curve in the use of DCNN-FuzzyWOA to detect COVID-19. Also, in order to be able to make a fair comparison, a simple DCNN has been used to detect COVID-19. This comparison is made because the test data set, the initial

values of the parameters, and the simple CNN structure, i.e., LeNet-5 DCNN, are entirely the same. According to what has been said, the competence and efficiency of DVNN-FuzzyWOA can be considered fair. On the test data set, the ROC curves demonstrate that DCNN-FuzzyWOA beats LeNet-5 DCNN considerably (Figure 4).

The suggested approach was implemented and executed 10 times, with a total training duration of between 4.5 and 11.5 minutes. The proposed classifier (DCNN-FuzzyWOA) for the COVID-19 validation set has a detection power between 99.01% and 100%. Due to the wide range of possible outcomes, the 10 trained DCNN-FuzzyWOA models are ensembled using weighted averaging with validation accuracy as the weights. The DCNN-FuzzyWOA classifier obtains a validation accuracy of 99.27 percent, while the LeNet-5 DCNN classifier achieves a detection accuracy of between 75.08 and 83.98 percent. The resultant ensemble achieves an 86.91 percent detection accuracy on the COVID-19 validation data set. New benchmark models including LeNet-5 DCNN [36], DCNN-GA [20], and DCNN-PSO [37] have been used to prove the efficiency and performance of DCNN-FuzzyWOA in detecting positive and negative cases of COVID-19. The ROC and precision-recall curves for the i-6c-2s-12c-2s and i-8c-2s-16c-2s structures are shown in Figures 5 and 6, respectively. The simulation results show that the DCNN-FuzzyWOA classifier or detector provides better results than other benchmark models.

For a more accurate comparison to understand the power and ability of DCNN-FuzzyWOA to detect positive and negative cases of COVID-19, more than 99.01% of the diagnoses are correct. The false alarm detection rate is less than 0.81%. In general, the trade-off between recall and precision for various threshold levels shows with the precision-recall curve. The greatest area under the precision-recall curve suggests that the accuracy and recall are strong. High precision shows a low false-positive rate, and high-

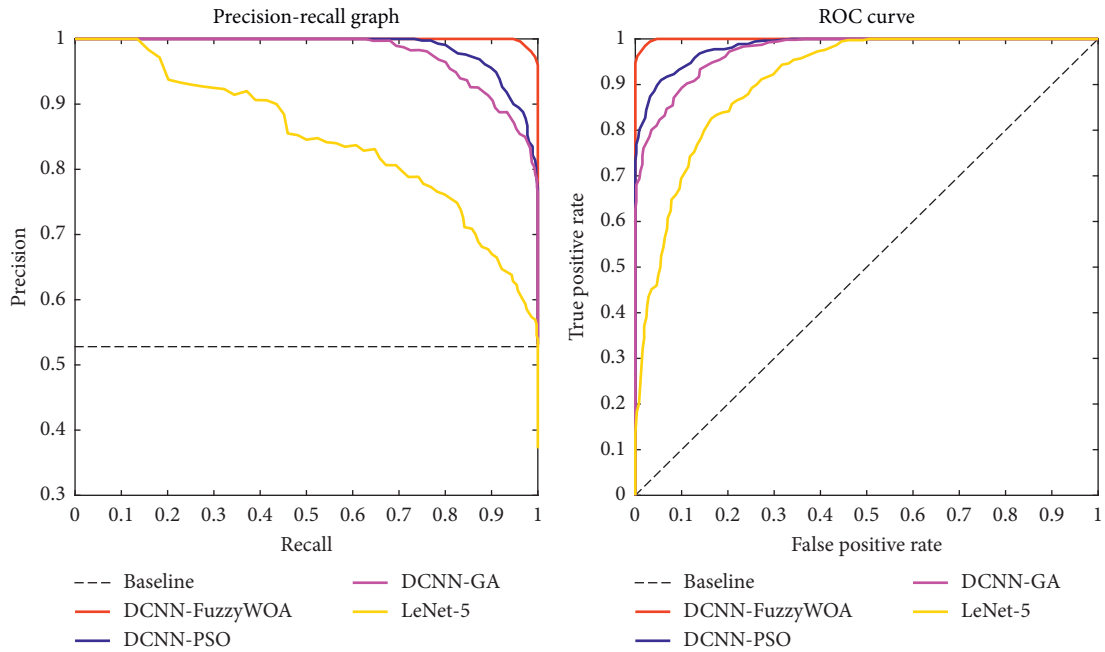


FIGURE 5: Curves of ROC and precision-recall for the i-6c-2s-12c-2s models.

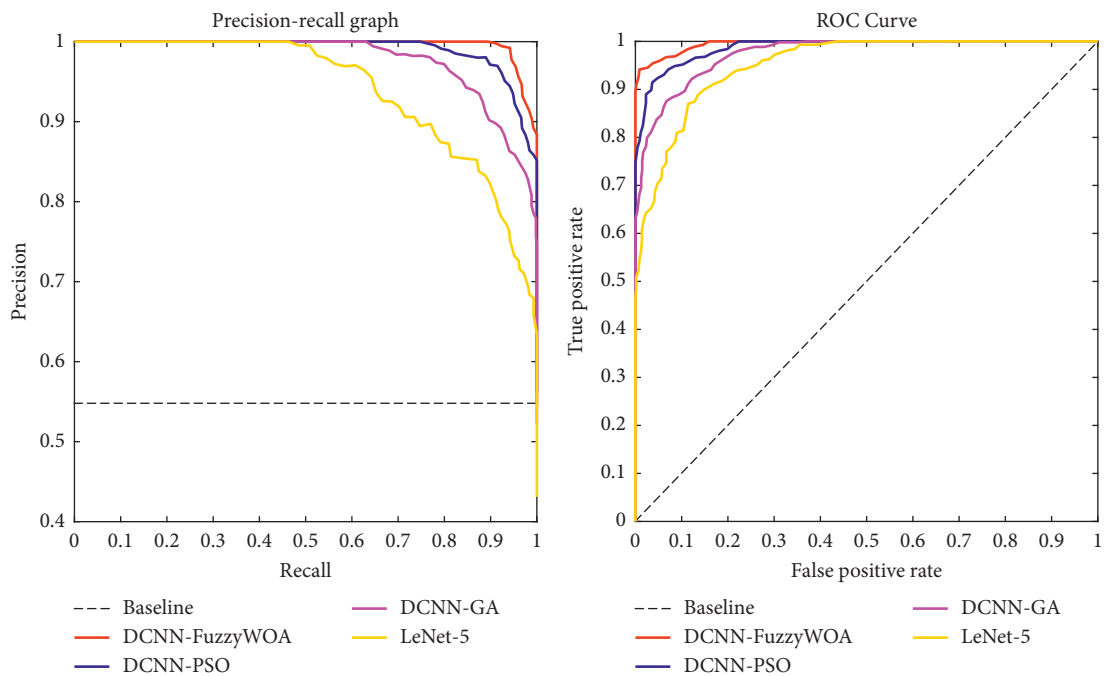


FIGURE 6: Curves of ROC and precision-recall for the i-8c-2s-16c-2s models.

recall indicates a low false-negative rate. Figures 5 and 6 show that DCNN-FuzzyWOA has the largest area under the precision-recall curve. It demonstrates a lower rate of false-positive- and false-negative classifications than other benchmark classifiers (see Tables 3–5).

Tables 3–6 describe the accuracy and computational time findings for the i-6c-2s-12c-2s and i-8c-2s-16c-2s structures. The overall result of the simulation was that the accuracy improved with increasing epoch. For example, in the first

epoch, compared to LeNet-5 (77.24), the accuracy increased to 3.84 for DCNN-GA (81.08), 8.63 to DCNN-PSO (89.71), and 1.73 for DCNN-FuzzyWOA (91.44). As shown in Table 3, the improvement in accuracy when 20 epochs are used is 1.57 for DCNN-GA (96.71), 2.05 for DCNN-PSO (98.76), and 1.24 for DCNN-FuzzyWOA (100). The simulation results show that DCNN-FuzzyWOA is more accurate in all epochs. As shown in Tables 4 and 6, processing time in FuzzyWOA is shorter and faster than other methods used.

TABLE 3: Accuracy and STD for the i-2s-6c-2s-12c structure.

Epoch	DCNN-FuzzyWOA		DCNN-PSO		DCNN-GA		LeNet-5	
	Accuracy	STD	Accuracy	STD	Accuracy	STD	Accuracy	STD
1	91.44	N/A	89.71	0.48	81.08	0.11	77.24	0.71
2	91.94	N/A	90.08	0.22	82.05	0.24	78.41	0.23
3	92.09	N/A	90.89	0.45	83.40	0.13	78.99	0.41
4	92.73	N/A	91.53	0.43	85.66	0.12	79.66	0.95
5	93.51	N/A	92.66	0.34	86.91	0.35	80.11	0.19
6	93.84	N/A	92.99	0.38	87.25	0.16	81.25	0.33
7	94.11	N/A	93.35	0.37	88.82	0.24	82.32	0.71
8	94.62	N/A	93.83	0.24	89.33	0.18	83.41	0.91
9	94.77	N/A	94.16	0.33	90.14	0.16	84.53	0.15
10	95.14	N/A	94.51	0.32	90.57	0.42	85.82	0.36
11	95.87	N/A	94.93	0.31	91.27	0.16	86.28	0.37
12	96.29	N/A	95.10	0.30	91.89	0.30	87.23	0.26
13	96.71	N/A	95.65	0.29	92.51	0.21	89.51	0.83
14	96.64	N/A	96.27	0.44	93.34	0.39	90.19	0.31
15	97.88	N/A	96.69	0.23	93.97	0.17	91.50	0.66
16	98.07	N/A	97.04	0.22	94.43	0.41	92.08	0.47
17	98.60	N/A	97.80	0.19	94.82	0.26	93.61	0.62
18	99.13	N/A	98.13	0.67	95.60	0.18	94.33	0.59
19	99.72	N/A	98.61	0.12	96.52	0.33	94.91	0.51
20	100	N/A	98.76	0.09	96.71	0.10	95.14	0.13

TABLE 4: Time required to compute and standard deviation for the i-2s-6c-2s-12c structure.

Epoch	DCNN-FuzzyWOA		DCNN-PSO		DCNN-GA		LeNet-5	
	Time	STD	Time	STD	Time	STD	Time	STD
1	85.91	N/A	108.55	1.04	115.01	0.78	127.08	0.81
2	115.87	N/A	199.43	1.02	161.76	1.71	195.20	1.07
3	184.65	N/A	283.71	2.08	221.95	2.41	238.85	2.58
4	222.41	N/A	305.86	1.07	260.74	1.09	299.50	1.17
5	291.33	N/A	390.29	1.23	317.55	4.99	310.17	4.37
6	301.96	N/A	448.91	2.11	361.34	3.14	422.39	1.08
7	345.17	N/A	519.57	1.56	433.98	2.08	531.81	2.09
8	379.86	N/A	589.39	1.84	549.27	1.19	579.27	4.01
9	405.16	N/A	618.28	2.42	625.10	1.78	536.90	1.28
10	476.22	N/A	697.68	3.86	677.31	2.77	640.33	4.65
11	495.57	N/A	737.70	3.07	731.79	1.18	678.88	2.65
12	511.79	N/A	793.32	1.73	792.03	3.34	723.74	1.59
13	577.73	N/A	836.15	1.66	841.50	4.28	791.83	2.66
14	601.63	N/A	889.04	2.37	881.53	3.11	845.70	2.13
15	647.85	N/A	923.17	2.09	903.72	1.56	936.62	1.83
16	690.33	N/A	978.64	1.88	930.18	4.66	1005.78	3.11
17	728.36	N/A	1001.79	3.77	982.04	1.23	1075.29	2.64
18	774.14	N/A	1060.8	1.91	1030.77	1.11	1103.21	2.23
19	834.71	N/A	1101.08	2.14	1161.20	3.28	1152.56	3.01
20	880.44	N/A	1186.61	1.89	1240.11	4.79	1256.07	1.74

As the number of epochs rises, the time efficiency of the FuzzyWOA becomes increasingly apparent, as the FuzzyWOA's stochastic structure results in a decrease in the complexity of the search space. It should be noted that the i-8c-2s-16c-2s structure findings in Tables 5 and 6

corroborate the previous conclusion about the i-8c-2s-16c-2s network. As a result, FuzzyWOA can significantly increase the performance of DCNNs with i-8c-2s-16c-2s and i-6c-2s-12c-2s structures. Data science experts believe that the best results can be shown using overall accuracy, ROC

TABLE 5: Accuracy and STD for the i-2s-8c-2s-16c structure.

Epoch	DCNN-FuzzyWOA		DCNN-PSO		DCNN-GA		LeNet-5	
	Accuracy	STD	Accuracy	STD	Accuracy	STD	Accuracy	STD
1	90.23	N/A	87.09	0.20	80.38	0.19	76.33	1.05
2	90.89	N/A	88.40	0.19	80.79	0.17	77.00	0.89
3	91.63	N/A	88.87	0.11	81.15	0.26	78.09	2.32
4	91.81	N/A	90.25	0.14	81.68	0.31	79.34	3.76
5	92.33	N/A	90.66	0.27	82.34	0.19	80.55	1.90
6	93.26	N/A	91.23	0.23	83.71	0.14	81.21	4.58
7	93.19	N/A	92.00	0.30	84.53	0.21	82.38	3.72
8	93.99	N/A	92.19	0.18	85.61	0.16	82.79	1.18
9	94.20	N/A	92.85	0.19	86.67	0.28	83.48	0.52
10	94.18	N/A	93.34	0.36	87.41	0.15	84.31	2.63
11	95.51	N/A	93.28	0.15	88.52	0.22	85.63	2.88
12	95.79	N/A	94.47	0.09	89.05	0.16	86.84	5.23
13	96.37	N/A	95.69	0.11	89.98	0.14	87.37	4.19
14	97.31	N/A	95.91	0.22	90.37	0.32	89.06	3.55
15	97.72	N/A	96.38	0.06	91.25	0.28	90.71	5.10
16	97.92	N/A	96.74	0.33	92.40	0.13	91.76	1.74
17	98.30	N/A	97.29	0.31	93.71	0.19	92.25	3.19
18	98.65	N/A	97.84	0.08	94.64	0.25	93.16	1.53
19	99.08	N/A	98.07	0.28	95.18	0.18	94.72	0.68
20	99.55	N/A	98.63	0.12	96.31	0.12	95.08	4.80

TABLE 6: Time required to compute and standard deviation for the the i-2s-8c-2s-16c structure.

Epoch	DCNN-FuzzyWOA		DCNN-PSO		DCNN-GA		LeNet-5	
	Time	STD	Time	STD	Time	STD	Time	STD
1	83.35	N/A	110.21	1.04	117.43	1.53	154.51	3.74
2	118.24	N/A	200.17	1.02	158.53	1.64	202.19	2.83
3	165.75	N/A	275.68	2.08	215.37	2.57	244.28	1.97
4	218.60	N/A	311.72	1.07	262.71	1.67	315.37	2.55
5	293.19	N/A	364.33	1.23	321.14	0.91	376.63	3.77
6	321.71	N/A	446.17	2.11	365.31	3.16	418.18	1.84
7	353.63	N/A	528.91	1.56	442.28	2.27	546.92	3.74
8	384.28	N/A	593.53	1.84	550.28	2.16	573.11	4.58
9	410.46	N/A	625.34	2.42	628.31	1.13	535.63	2.63
10	496.39	N/A	670.81	3.86	680.32	4.28	632.27	0.63
11	508.77	N/A	741.73	3.07	734.62	5.33	689.81	3.27
12	542.91	N/A	799.84	1.73	783.49	2.59	722.35	3.36
13	596.72	N/A	842.59	1.69	853.78	1.49	793.44	1.25
14	663.85	N/A	891.70	2.37	892.75	2.27	835.23	2.80
15	689.51	N/A	928.91	2.08	913.36	1.56	947.95	2.33
16	734.38	N/A	974.32	1.87	936.77	2.23	1025.52	4.20
17	770.41	N/A	1011.30	3.76	980.19	1.44	1098.37	0.76
18	829.13	N/A	1063.85	1.95	1032.83	1.78	1110.50	1.58
19	857.67	N/A	1127.63	2.32	1163.27	2.56	1153.48	0.99
20	945.61	N/A	1201.21	1.89	1262.46	5.11	1398.13	1.81

TABLE 7: Comparison of F1-Score in structures i-2s-6c-2s-12c and i-2s-8c-2s-16c.

Structure	i-2s-6c-2s-12c				i-2s-8c-2s-16c			
	F1-score %				F1-score %			
Epoch	DCNN-FuzzyWOA	DCNN-PSO	DCNN-GA	LeNet-5	DCNN-FuzzyWOA	DCNN-PSO	DCNN-GA	LeNet-5
1	89.10	89.71	73.21	70.06	89.89	80.51	77.84	0.10
2	89.13	87.86	73.87	78.41	90.25	81.19	78.45	0.42
3	89.89	89.04	75.62	73.24	90.63	82.27	79.86	1.53
4	90.78	90.18	75.75	74.93	91.03	83.97	81.68	2.46
5	91.20	92.14	77.19	75.08	91.19	84.23	83.54	1.90
6	91.22	92.31	77.73	75.62	91.43	84.35	84.94	2.68
7	91.49	92.45	78.62	76.80	93.19	92.00	85.53	1.52
8	91.98	92.45	79.79	76.71	92.44	86.54	86.54	1.18
9	93.44	92.83	80.81	79.82	94.08	86.11	87.68	0.52
10	93.47	93.10	82.11	80.74	94.02	88.37	87.41	1.27
11	94.19	93.13	85.51	80.97	94.51	88.62	88.52	1.36
12	94.28	93.27	91.89	83.44	95.79	89.49	89.18	1.89
13	95.31	93.76	88.48	87.32	96.37	90.13	89.57	2.25
14	95.77	95.00	93.35	88.18	97.31	91.91	89.96	1.46
15	97.01	95.13	89.22	89.04	95.11	92.81	90.14	3.91
16	97.47	95.50	90.67	89.49	95.34	92.65	90.72	1.23
17	97.90	95.89	91.34	89.78	95.89	94.29	93.71	2.85
18	98.53	96.31	91.52	90.08	96.51	95.81	94.64	0.79
19	98.99	96.66	96.25	91.37	97.00	96.07	94.89	0.91
20	100	96.97	93.15	91.45	97.31	96.73	94.32	2.89

curve, F1-Score. Therefore, Table 7 examines the F1-Score in structures i-2s-6c-2s-12c and i-2s-8c-2s-16c.

As shown in Table 7, the results obtained from FuzzyWOA are more appropriate and encouraging than the other methods used. So that, in the twentieth epoch, in the structure of i-2s-6c-2s-12c, the value of F1-Score reaches 100%.

#### 4. Conclusion

In this paper, using AI tools, i.e., a combination of DCNN, WOA, and fuzzy logic, an accurate model is designed and proposed to detect the positive and negative cases of COVID-19 using X-ray. In addition to using the COVID-Xray-5k benchmark data set, the DCNN-PSO, DCNN-GA, and DCNN classic models were used for a fair comparison of the proposed detector or classifier. Analysis of simulation results provided comparable and significant results for the proposed DCNN-FuzzyWOA model. Experts also confirmed the relationship between the results and clinical results. One of the most significant reasons for the optimal performance of the DCNN-FuzzyWOA model is the adjustment of WOA control parameters by the fuzzy system and the determination of a clear boundary between the exploration and extraction phases in the search space of the WOA trainer algorithm. All training algorithms used to train the two convolutional networks were compared in terms of accuracy, processing time, F1-Score, and curves of ROC and precision-recall. The results showed that FuzzyWOA had a more encouraging performance than the other methods used. In terms of structure, the i-2s-6c-2s-12c architecture has been more successful. Of course, despite getting good results from DCNN-FuzzyWOA, larger data sets than COVID-19 are needed to achieve higher accuracy with more excellent reliability.

#### Data Availability

Data are available and can be provided over the emails querying directly to the author at the corresponding author (abbas.saffari@birjand.ac.ir).

#### Conflicts of Interest

The authors declare that they have no conflicts of interest.






#### References

- [1] P. Luz, E. Silva, G. Moreira et al., "COVID-19 detection in CT images with deep learning: a voting-based scheme and cross-datasets analysis," *Informatics in Medicine Unlocked*, vol. 20, Article ID 100427, 2020.
- [2] T. Yang, Z. Hou, H. Zhan et al., "Correlation of chest CT and RT-PCR testing for coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases," *Radiology*, vol. 296, no. 2, pp. E32–E40, 2020.
- [3] A. Sharma, S. Rani, and D. Gupta, "Artificial intelligence-based classification of chest X-ray images into COVID-19 and other infectious diseases," *International Journal of Biomedical Imaging*, vol. 2020, pp. 1–10, 2020.
- [4] T. First, "Handbook of COVID-19 Prevention and Treatment,".
- [5] R. Vaishya, M. Javaid, I. H. Khan, and A. Haleem, "Artificial Intelligence (AI) applications for COVID-19 pandemic," *Diabetes & Metabolic Syndrome: Clinical Research Reviews*, vol. 14, no. 4, pp. 337–339, 2020.
- [6] A. Abdelsamea, M. M. Gaber, and M. M. Gaber, "DeTrac: transfer learning of class decomposed medical images in convolutional neural networks," *IEEE Access*, vol. 8, pp. 74901–74913, 2020.
- [7] S. Bhattacharya, P. K. Reddy Maddikunta, Q. V. Pham et al., "Deep learning and medical image processing for coronavirus

- (COVID-19) pandemic: a survey,” *Sustainable Cities and Society*, vol. 65, Article ID 102589, 2021.
- [8] Q. V. Le, A. Coates, B. Prochnow, and A. Y. Ng, “On Optimization Methods for Deep Learning,” 2011.
- [9] D. Povey, “Krylov Subspace Descent for Deep Learning,” 2015, <https://arxiv.org/abs/1111.4259>.
- [10] J. Martens, *Deep Learning via Hessian-free Optimization*, 2010.
- [11] A. J. Holden, “Reducing the Dimensionality of,” vol. 313, pp. 504–507, 2006.
- [12] M. Mosavi and M. R. Mosavi, “Chimp optimization algorithm,” *Expert Systems with Applications*, vol. 149, Article ID 113338, 2020.
- [13] S. Arora, S. Singh, and K. Yetilmezsoy, “A modified butterfly optimization algorithm for mechanical design optimization problems,” *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 40, no. 1, p. 21, 2018.
- [14] A. Saffari, S. H. C. A. Zahiri, and M. Khishe, “Fuzzy whale optimisation algorithm: a new hybrid approach for automatic sonar target recognition,” *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 18, no. 1, pp. 1–17, 2022.
- [15] Y. Zhining and P. Yunming, *The Genetic Convolutional Neural Network Model Based on Random Sample*, vol. 8, no. 11, pp. 317–326, 2015.
- [16] G. Rosa, “Fine-tuning convolutional neural networks using Harmony search fine-tuning convolutional neural networks using Harmony search,” no., June, 2016.
- [17] M. Irfan, M. A. Iftikhar, S. Yasin et al., “Role of hybrid deep neural networks (HDNNs), computed tomography, and chest X-rays for the detection of COVID-19,” *International Journal of Environmental Research and Public Health*, vol. 18, no. 6, p. 3056, 2021.
- [18] Y. E. Almalki, A. Qayyum, M. Irfan et al., Alkhalik, A. Alduraibi, M. Saeed, and S. Rahman, A novel method for COVID-19 diagnosis using artificial intelligence in chest X-ray images,” *Healthcare*, vol. 9, no. 5, p. 522, 2021.
- [19] H. Fan, L. Zheng, and Y. Yang, “Unsupervised Person Re-identification,” pp. 1–9, 2018.
- [20] Y. Sun, B. Xue, M. Zhang, G. Yen, and J. Lv, “Automatically designing CNN architectures using the genetic algorithm for image classification,” *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 3840–3854, 2020.
- [21] Y. Wang, L. P. Yuan, M. Khishe, A. Moridi, and F. Mohammadzade, “Training RBF NN using sine-cosine algorithm for sonar target classification,” *Archives of Acoustics*, vol. 45, no. 4, pp. 753–764, 2020.
- [22] M. Mohammadi and H. Mohammadi, “Passive sonar target classification using multi-layer perceptron trained by salp swarm algorithm,” *Ocean Engineering*, vol. 181, pp. 98–108, 2019.
- [23] M. R. Mosavi, M. Khishe, M. J. Naseri, G. R. Parvizi, and M. Ayat, “Multi-layer perceptron neural network utilizing adaptive best-mass gravitational search algorithm to classify sonar dataset,” *Archives of Acoustics*, vol. 44, no. 1, pp. 137–151, 2019.
- [24] P. S. Optimization, “Comparison of particle swarm optimization and backpropagation as training,” *Algorithms for Neural Networks*, vol. 21 page.
- [25] M. Kaveh, M. Khishe, and M. R. Mosavi, “Design and implementation of a neighborhood search biogeography-based optimization trainer for classifying sonar dataset using multi-layer perceptron neural network,” *Analog Integrated Circuits and Signal Processing*, vol. 100, no. 2, pp. 405–428, 2018.
- [26] M. Safari and A. Safari, “Classification of sonar targets using an MLP neural network trained by dragonfly algorithm,” *Wireless Personal Communications*, vol. 108, no. 4, pp. 2241–2260, 2019.
- [27] A. Saffari, M. Khishe, and S. H. Zahiri, “Fuzzy-ChOA: an improved chimp optimization algorithm for marine mammal classification using artificial neural network,” *Analog Integrated Circuits and Signal Processing*, vol. 111, no. 3, pp. 403–417, 2022.
- [28] S. Lewis and A. Lewis, “The whale optimization algorithm,” *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.
- [29] A. Saffari and M. Khishe, “Classification of marine mammals using trained multilayer perceptron neural network with whale algorithm developed with fuzzy system,” *Preprint in Research Square*, 2020.
- [30] M. Xin and Y. Wang, “Research on image classification model based on deep convolution neural network,” vol. 8, 2019.
- [31] M. I. Uddin, S. Atif, A. Shah, and M. A. Al-khasawneh, “Research Article A Novel Deep Convolutional Neural Network Model to Monitor People following Guidelines to Avoid COVID-19,” vol. 2020, Article ID 8856801, 2020.
- [32] A. Khan, A. Sohail, U. Zahoor, and A. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5455–5516, 2020.
- [33] S. Minaee, R. Kafieh, M. Sonka, S. Yazdani, and G. Jamalipour Soufi, “Deep-COVID: predicting COVID-19 from chest X-ray images using deep transfer learning,” *Medical Image Analysis*, vol. 65, Article ID 101794, 2020.
- [34] J. P. Cohen, P. Morrison, and K. Roth, “COVID-19 Image Data Collection: Prospective Predictions Are the Future,” pp. 1–38, 2019.
- [35] J. Irvin, P. Rajpurkar, M. Ko et al., “with Uncertainty Labels and Expert Comparison,” vol. 33, 2019.
- [36] H. Berg and K. T. Hjelmervik, “Classification of anti-submarine warfare sonar targets using a deep neural network,” *Ocean*, pp. 1–5, 2018.
- [37] B. Wang, B. Xue, and M. Zhang, “Particle Swarm Optimization for Evolving Deep Neural Networks for Image Classification by Evolving and Stacking Transferable Blocks,” in *Proceedings of the 2020 IEEE Congress on Evolutionary Computation (CEC)*, Glasgow, UK, July 2020.

## Research Article

# GC-CNNnet: Diagnosis of Alzheimer's Disease with PET Images Using Genetic and Convolutional Neural Network

Morteza Amini <sup>1</sup>, Mir Mohsen Pedram <sup>1,2</sup>, AliReza Moradi <sup>3,4</sup>, Mahdieh Jamshidi <sup>1</sup>, and Mahshad Ouchani <sup>4</sup>

<sup>1</sup>Department of Cognitive Modeling, Institute for Cognitive Science Studies, Tehran, Iran

<sup>2</sup>Department of Electrical and Computer Engineering, Faculty of Engineering, Kharazmi University, Tehran, Iran

<sup>3</sup>Department of Clinical Psychology, Faculty of Psychology and Educational Science, Kharazmi University, Tehran, Iran

<sup>4</sup>Department of Cognitive Psychology, Institute for Cognitive Science Studies, Tehran, Iran

Correspondence should be addressed to Morteza Amini; [amini\\_m@icss.ac.ir](mailto:amini_m@icss.ac.ir) and Mir Mohsen Pedram; [pedram@khu.ac.ir](mailto:pedram@khu.ac.ir)

Received 8 December 2021; Revised 1 June 2022; Accepted 10 June 2022; Published 9 August 2022

Academic Editor: Laith Abualigah

Copyright © 2022 Morteza Amini et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

There is a wide variety of effects of Alzheimer's disease (AD), a neurodegenerative disease that can lead to cognitive decline, deterioration of daily life, and behavioral and psychological changes. A polymorphism of the ApoE gene  $\epsilon$  4 is considered a genetic risk factor for Alzheimer's disease. The purpose of this paper is to demonstrate that single-nucleotide polymorphic markers (SNPs) have a causal relationship with quantitative PET imaging traits. Additionally, the classification of AD is based on the frequency of brain tissue variations in PET images using a combination of  $k$ -nearest-neighbor (KNN), support vector machine (SVM), linear discrimination analysis (LDA), and convolutional neural network (CNN) techniques. According to the results, the suggested SNPs appear to be associated with quantitative traits more strongly than the SNPs in the ApoE genes. Regarding the classification result, the highest accuracy is obtained by the CNN with 91.1%. These results indicate that the KNN and CNN methods are beneficial in diagnosing AD. Nevertheless, the LDA and SVM are demonstrated with a lower level of accuracy.

## 1. Introduction

AD is defined by irregular extracellular  $\beta$ -amyloid plaques and intraneuronal tau aggregation on a neuropathological level (neurofibrillary tangles). The concept of an AD continuum, which contains both typical and atypical manifestations of the disease, arose from observations that patients with various clinical appearances and progressions have identical neuropathological features [1, 2]. AD is a late-onset condition in more than 80% of patients (defined haphazardly as cases with 65 years or older). Mild cognitive impairment (MCI) is a dementia prodromal phase that affects the voice, visuospatial, praxis, and executive domains and worsens over time. On the other hand, patients with early-onset AD usually present with a more severe multidomain cognitive disorder impacting memory, concentration,

vocabulary, visuospatial, and executive functions at the time of diagnosis. In patients with early-onset AD, except for the elderly, MCI rarely precedes primary cognitive dysfunction, which also develops more rapidly to severe steps. As a result, in 2010, 2011 [3, 4], and 2014, atypical AD variations were applied to the updated diagnosis guidelines for AD. The most recent updated form [5] involves (1) a clinical phenotype associated with one of the atypical forms of dominant, progressive, frontal and (6) logopenic variant, visual/posterior variant and (2) biochemical, genetic, and/or in vivo molecular imaging symptoms confirming AD diagnosis. Nevertheless, new clinical phenotypes of AD have been recorded recently in patients with semantic variant predominantly progressive aphasia [6] or corticobasal syndrome [7]. These innovative clinical variations add to the taxonomy of AD, accentuate a broad range of patient

features. Most notably, the diagnosis of early-onset variants of Alzheimer's poses critical challenges [8] underlines the significance of biomarkers for detection in vivo.

FDG-PET is a promising modality for forecasting adaptive brain alterations in AD, detecting variations early in the disease, and recognizing AD from other dementias. Several studies on the efficacy of FDG-PET in AD have been published over the last three decades. A meta-analysis of 27 FDG-PET investigations in the diagnosis of AD finds a 91% (95% confidence interval, 86%–94%) and 86% (95% CI, 79%–91%) pooled sensitivity, and 86% (95% CI). The study included 119 papers examining the function of different diagnostic methods in AD. The meta-analyzes find that FDG-PET has outstanding diagnostic accuracy [9, 10] compared to other diagnostic approaches such as clinical guidance, MRI, CT, SPECT, and biomarkers. Besides, tests have shown that FDG-PET can differentiate patients with AD from stable controls and dementia from other diseases. The FDG-PET is to recognize 98 and 99 percent AD patients with normal SN and SP subjects, 99 and 98 percent DLB with SN and SP patients with 99% and 71 percent SN and SP and 99 and 65 percent FTD patients with SN and SP, according to Mosconi et al. [11]. Neuroimaging has been critical in supporting underlying pathophysiological hypotheses regarding the condition over the past two decades, and it has primarily been linked to the evolution of diagnostic methods. The conditions for amnesic (typical) AD causes have been revised: MRI hippocampal atrophy, temporoparietal hypometabolism of FDG-PET, and elevated fibrillar amyloid PET-amyloid accumulation in the brain. In particular, exposure to imaging biomarkers raises the likelihood of an AD diagnosis even under preclinical/pre-dementia circumstances [5].

Is molecular PET imagery able to explain the phenotypic variety of AD and explain whether and how pathologic  $\beta$ -amyloid and tau proteins show the clinical appearance of the disease? To date, amyloid PET tests have seen diffuse cortical  $\beta$ -amyloid deposits in patients with average or atypical early-onset AD, irrespective of clinical presentation. The relationship between cognitive profile, metabolic transition, and irregular protein distribution has been small. Furthermore, this radiotracer family has not shown a distinct geographic trend between focal and diffuse AD [12, 13]. On the other hand, PET tracers that target tau have revealed a close link between tau deposit distribution and clinical phenotype.

For the purposes of monitoring the progression of AD, we examined genes that have significant correlations with statistical properties of three PET tracers other than the ApoE genotype. In this article, 37 characteristics are discussed to assist in diagnosing Alzheimer's disease. PET images provide inputs for different parts of the brain depending on their frequency dependence. Nearest-neighbor (KNN), support vector machine (SVM), linear discrimination analysis (LDA), and convolutional neural network (CNN) are four machine learning approaches used to diagnose Alzheimer's disease. Several reduced features are used to create the input layer, while two MCI labels and the normal value are used to construct the output layer.

The following sections of the present paper outline: Section 1 describes medical imaging and key medical imaging characteristics and quality factors. Section 2 reviews many relevant papers in medical image processing and studies some image processing methods for improving medical images that researchers have proposed in their papers. Section 3 is the core of the present research paper. This section explains some of the significant engineering subjects related to image processing, general, and medical imaging, particularly in Section 4. The evaluation metrics are discussed in Section 5. Finally, Section 6 summarized the numerical results and future works.

## 2. Literature Review

Alzheimer's disease is a neurodegenerative disease with distinct pathologic characteristics. Although cortical and hippocampal neuronal dysfunction and generalized gray matter atrophy are hallmarks of Alzheimer's disease, patients can also experience gradual disconnection of cortical and subcortical regions attributable to white matter injury. AD is a progressive disease that worsens over time. The ApoE genotype  $\epsilon 4$  is well known as a genetic risk factor for AD. Furthermore, PET/MRI is a systematic instrument for clinical detection of AD by identifying changes in the brain. We looked at single-nucleotide polymorphisms (SNPs) focused on whole-genome sequencing (WGS) data in this research.

The biochemical structures found with the gold standard of PET imaging of fluorodeoxyglucose (FDG) strongly mimic the cortical distribution of tau protein: hypometabolism is a pathologically intimate result of tau deposition [13, 14]. In short, in the early-onset Alzheimer varieties, the function and density of tau aggregation are locally linked to cognitive effects, cerebral blood pressure, atrophy, and metabolic changes, while  $\beta$ -amyloid is diffusing [13]. The area of study in brain imaging genetics explores the effect of genetic variations on brain imaging phenotypes. It examines how genetic variations such as single-nucleotide polymorphisms (SNPs) and quantitative traits (QTs) derived from brain imaging evidence contribute to phenotypical features and molecular mechanics in complicated brain conditions. Single voxels [14] or regions of interest (ROIs) [15–17] in the brain are used to calculate imaging QTs. An ROI is a predetermined brain region consisting of an anatomical/functionally annotation similar cluster of voxels. The ROI number (ten hundred) in the cortex is significantly smaller than the voxel number (tens of thousands to many millions).

Recent advancements in obtaining multimodal neuroimaging technology inherently have precise voxel-level knowledge, which opens up a plethora of possibilities for investigating fine-grained brain anomalies. Voxelwise methods to investigate genetic implications for voxel-based brain measures have been suggested in brain imaging genetics. Stein et al. [14] suggested that GWAS (vGWAS) could be included in an AD analysis to evaluate relationships paired by 448,293 SNPs and 31,622 voxels. Hibar et al. [18] proposed the voxelwise gene-wide interaction study



(vGeneWAS), which compared the combined influence of several SNPs within a gene to voxel-level measures using a multivariate model. In their study, He et al. [19] studied several methods for selecting data features to achieve dimensionality reduction. Chen et al. [20] developed Deep-M6ASeq-EL, which utilizes an ensemble of five LSTMs and CNNs with a hard voting strategy. According to Xu et al. [21], pathogenesis can be represented using a directed graph (PN) in a heuristic way.

When vGWAS (e.g., CSMD2 and CADPS2) and vGeneWAS (e.g., GAB2) have detected specific genes, no primary genetic imaging links have been found have survived the correction of several tests. In organized sparse learning, Du et al. [40] proposed two new penalties to strengthen the fused lasso and the graph/network-driven lasso penalties. They penalized the SCCA model in both ways and proposed an optimization algorithm to solve it. The suggested SCCA approach had a clear upper limit on grouping results positively and negatively correlated variables. In discovering biologically significant imaging genetic associations, the suggested technique detected higher canonical correlation coefficients and captured simpler canonical weight patterns. Auditory verbal learning test delayed recall (AVLT-DR) regressing 6-month AVLT-DR (AD neuroimaging Initiative database) scores in 394 individuals with adequate knowledge at baseline AVLT-DR scores. According to the findings, loss of practice effect over six months can be as effective as biomarkers in predicting 6-year AD risk.

The study by Yao et al. [41] proposed voxel-wise enrichment analysis that integrates brain-anatomic annotation results as an efficient and robust means for mining regionally based imaging genetic associations recognizing the mutual impact of weak voxel-level signals. In order to investigate the genetic effects of imaging on the brain, the proposed technique has demonstrated to be both scalable and effective. In a study by Zhao et al. [42], the Multiple Kernel-based Fuzzy SVM Model with Support Vector Data Description (MK-FSVM-SVDD) was proposed in order to predict DBPs. In Yan et al. [43], plasma-activated water (PAW) and heat-moisture treatments (HMT) were combined to study the structure, physical properties, and in vitro digestibility of waxy (WMS) and normal maize starches (NMS). In Shi et al. [44], the effect of WSG and its impact on steamed bread quality were studied. Increasing ultrasonic intensity first increased and then decreased the complex index (CI) of the WSG. Nejatishahidin et al. [45] developed a novel pose estimation model for object categories that can be effectively applied to previously unknown environments. Eslami et al. [46] showed that attention-based multiscale convolutional neural networks (A+MCNNs) could improve the automated detection of common distress and nondistress objects in pavement images. In this study, Dubois et al. [47] investigated epigenetic processes as they relate to psychiatric disorders and traumatic or stressful events, family relationships, and also gut microbiota. Wang et al. [48] used the BP neural network algorithm to train the input value of the network marketing and to judge the risk. Prasad et al. [49] used response surface methodology (RSM) and artificial neural

network (ANN) to predict the color removal by adsorption. Rezaei et al. [50] introduced a data-driven method to segment hand parts from depth maps without requiring any additional effort to obtain segmentation labels. In their study, Chandra et al. [51] examined in vivo molecular imaging in relation to amyloid, tau, and microglial activation in AD pathology. As part of the study, PET imaging tests were examined as possible biomarkers and ways to control disease development (see Table 1). In recent research, metaheuristic optimization methods have grown more attractive [52, 53]. Because they can solve multiple-objective solutions and nonlinear formulations, metaheuristics are increasingly being utilized to find high-quality solutions to a growing number of complex real-world problems [54–58]. Optimization approaches underpin a wide range of essential tasks, and they may be used to solve a wide range of image segmentation issues in medicine [59–63]. In summary, imaging genetics investigation focuses on ROI-level phenotypes such as (i) low dimensionality relative to voxel-based computational strength approaches and (ii) structural or functional ROI annotations to indefinite analysis. AD is a progressive disease that worsens over time. The ApoE genotype  $\epsilon 4$  is well known as a genetic risk factor for AD. Furthermore, PET/MRI is a systematic instrument for clinical detection of AD by identifying changes in the brain. We looked at single-nucleotide polymorphisms (SNPs) focused on whole-genome sequencing (WGS) data in this research. We discovered several SNPs that have a strong link to PET imaging quantitative traits (QTs). Moreover, the classification is done to diagnose AD based on the frequency of different brain parts in PET images. Analysis metrics are used to illustrate the results. Machine learning is also widely used in biological applications, such as optimization [63, 64], feature extraction [65, 66], and diagnosis of tumors [67]. The applications of deep learning method are infection disease detection [68], economical application [69], cancer research [70], brain tumor detection [71, 72], fatigue detection [73], environmental science [74], federated learning [75], facial expression detection [76], and healthcare analysis [77]. Moreover, some metaheuristic methods are aquila optimization [78], reptile search method [79], genetic algorithm [74], and so on [80].

### 3. Methods and Materials

**3.1. PET Imaging Genetics.** PET imaging genetic expression can be precisely accomplished by radiolabeling samples that only bind certain parts of the target molecule (e.g., protein, mRNA, or DNA) or radiolabeling samples, which are explicitly metabolized by a particular enzyme or sequence of reactions leading to a radiolabeling complex that is “trapped” in the tissue. There are also instances of the nuclear medicine direct imaging model. In various areas, including neuroscience studies, PET imagery of receptor density/occupancy with little radio-labeled molecular sensors is widely used. Another instance of direct molecular imaging that has progressed over the last 30 years [81] is picking cell surface-specific antigens or epitopes with radiation-labeled antibodies.

TABLE 1: The literature reviews.

Ref	Probe	Results
[22]	[ <sup>11</sup> C]PBB3	[ <sup>11</sup> C] PBB3 was substantially higher in Alzheimer's disease than controls in medial temporal areas, including the hippocampus
[23]	[ <sup>11</sup> C]PBB3	In neocortical regions, particularly the medial temporal-co, significant variations in tracer uptake were found, while the Alzheimer's disease spectrum was comparable to normal controls. The group also experienced MRI medial time atrophy. Besides, the intake of cognitive status in front and temporoparietal joints, limbic, paralimbic, and frontoparietal zones, was positively linked with dementia, and frontal uptake of Alzheimer's patients in frontal regions was also correlated positively with frontal executive dysfunction
[24]	[ <sup>11</sup> C]PBB3 [ <sup>11</sup> C] THK5351	[ <sup>11</sup> C]THK5351 displayed larger percent in the temporal lobe of the medium and lateral lobe, and the reverse was shown in a combination of patients of Alzheimer's disease and mild cognitive impairment. [ <sup>11</sup> C]PBB3 is implicated in the uptake of PET amyloid. The brain uptake of [ <sup>11</sup> C]THK5351 and [ <sup>11</sup> C]PBB3 has shown to be adversely linked to cognitive efficiency
[25]	[ <sup>18</sup> F] THK5317	The lat-temporal, lat-occipital-, inf-parietal, anterior, lat-occipital-co, and precuneus patients with mild cognitive impairment and Alzheimer's disease have greater tau connection than in healthy individuals. In PET, tau retention and fluorodeoxyglucose uptake were harmful in the frontal-Co, but the tau and the amyloid bonding were positive in the neocortex
[26]	[ <sup>18</sup> F] THK5351	As contrasted to healthy controls, the eroded WM, fusiform gyrus, inf-temporal-co, lingual gyrus, mid-temporal gyrus, occipital-Co, parietal-Co, post-cingulate, and precuneus all indicated increment tracer absorption
[27]	[ <sup>18</sup> F] THK5317	The occipital regions, the mid-frontal and post-cingulate gyri, the parietal operculum, the precuneus, and the parahippocampal, fusiform, intermediate, lower, and superior temporal gyri, were observed to be adversely linked to memory in Alzheimer's patients. Fluorodeoxyglucose-PET studies, which revealed an essential correlation between tau binding and cognition, affected the impact of in vivo tau binding on cognition
[28]	[ <sup>18</sup> F] THK5351 [ <sup>18</sup> F]AV-1451	Uptake of [ <sup>18</sup> F]THK5351 was greater in Alzheimer's patients in the cerebral temporal and occipital regions than in healthy controls; in the hippocampus, [ <sup>18</sup> F]AV1451 uptake was higher
[29]	[ <sup>18</sup> F]AV-1451	In all four lobes of the cortex as well as of the hippocampus, the connections with Alzheimer's disease were more robust in comparison with stable controls
[30]	[ <sup>18</sup> F]AV-1451	In Alzheimer's disease patients in hippocampal and extensive cortical areas, tracer retention was more remarkable compared to control
[31]	[ <sup>18</sup> F]AV-1451	A significant proportion of cortical regions examined in Alzheimer's disease have greater tau uptake than controls. This condition persisted in mild cognitive impairment for the entorhinal-Co
[33]	[ <sup>18</sup> F]AV-1451	The cortical preservation of [ <sup>18</sup> F]AV1451 was higher than the controls for the temporoparietal, parietooccipital, precuneus post-cingulate, and frontal areas in mixed patient groups. In the entorhinal, parahippocampal, inferior temporal, and fusiform-Co also variations were reported. Cognitive impairment and dementia severe were associated with increased inferior uptake for patients
[34]	[ <sup>18</sup> F]AV-1451	The frontal, occipital, parietal, and temporal-co, as well as the amygdala, anterior and post-parahippocampus, and fusiform areas, displayed elevated levels of tau binding relative to controls in the frontal, occipital, parietal, and temporal-co, as well as the amygdala, anterior and post-parahippocampus, and fusiform sections of Alzheimer's disease and mild cognitive impairment patients
[35]	[ <sup>18</sup> F]AV-1451	Variation of entorhinal and neocortical tau binding was observed in patients with classic Alzheimer's disease. The tremendous memory damage being found by people with higher entorhinal and neocortical tracer retention, while those with low entorhinal and elevated neocortical attachment were the most deteriorating in other areas of neuropsychology, according to a cluster study contrasting high and low uptake groups
[36]	[ <sup>18</sup> F] THK5317	In Alzheimer's disease patients, in addition to the midbrain, [ <sup>18</sup> F]THK5317 binding was found in basal ganglia and thalamus. The isocratic temporal lobe and lateral parietal and frontal lobes retention were observed in the tracer retention
[37]	[ <sup>18</sup> F]MK-6240	In the medial temporal lobe, both amygdala, hippocampus, and parahippocampal gyrus demonstrated increased tracer uptake in patients with AS/Mild cognitive impairment. In the neocortical temporal, frontal, and parietal regions, two patients with progressive disease were taken up
[38]	[ <sup>18</sup> F]PI-2620	In the temporal areas, the precuneus, and the post cingulate, three Alzheimer's disease patients had asymmetric distributions of tracer retention. One Alzheimer's disease patient, who was in the early stages of the disorder, <i>h</i> Alzheimer's disease little absorption
[39]	[ <sup>18</sup> F]RO-948	Alzheimer's disease patients had higher tracer attachment than older controls in the right hippocampus, entorhinal area, parahippocampus, left middle-middle front lobe, fusiform gyrus, mid temporal-Co, inferior lobe, and right inferior parietal lobe
[40]	[ <sup>18</sup> F]GTP1	Braak stage I/II brain regions have better retention of tracer in mild to moderate Alzheimer's disease patients than CN brain regions, and braak stage V/VI brain regions have higher retention of tracer

**3.2. Convolutional Neural Network.** A CNN is a deep learning (DL) method that can take an input matrix and assign importance (learnable weights and biases) to different aspects/objects while also distinguishing between them. In comparison to other classification methods, a CNN requires significantly less preprocessing. In spite of the rudimentary design of filters, CNN can learn these filters/characteristics with enough training. CNN architecture was inspired by the structure of the visual cortex, which is similar to the pattern of communication between neurons in the human brain. Individual neurons are only capable of responding to stimuli that are located within the receptive field, a small portion of the visual field. When multiple such fields collide, the entire visual field becomes occupied [82]. Manual attribute extraction methods, including such texture analysis, are used in the majority of recent radionics experiments, accompanied by traditional machine learning (ML) methods, like random forests and support vector machines (SVM) [83]. There are a few distinctions to be made between those approaches and CNN. To begin with, CNN does not necessitate feature extraction by hand. Second, human experts are rarely used to segment tumors or organs in CNN architectures. Third, because millions of learning parameters are necessary to predict, CNN is much more data hungry and computer intensive, and GPUs are required for model training. Among the building blocks of CNN architecture are convolution layers, pooling layers, and fully connected layers. One or two fully connected layers follow a stack of multiple convolution layers and a pooling layer in the typical architecture. The way the input data is converted to output data that uses these layers is called forward propagation. Though 2D-CNN is used for convolution and pooling, the associated three-dimensional (3D)-CNN operations can also be applied [84].

**3.3. Support Vector Machine (SVM).** SVM is the most used (ML)-based pattern classification technique today. It was created by Vapnik in 1995 and is centered on mathematical learning theory. The main goal of this methodology is to use various types of kernel functions to project nonlinearly separable samples onto a higher dimensional space. Kernel methods have gotten much attention recently, thanks to the growing success of SVM [85]. Kernel functions are essential in SVM for bridging the gap between linearity and non-linearity. The least-square SVM technique is another helpful SVM methodology for classification tasks. For grouping, extreme learning machines, fuzzy SVMs, and genetic algorithm-tuned expert models can all be used. Three different kernel functions, namely linear, polynomial, and RBF kernels, were tested in this analytical work [85].

**3.4. K-Nearest-Neighbor Classifier (KNN).** The KNN classifier is a common and useful data mining tool. KNN classifies each test sample based on its  $k$  nearest neighbors. The distance between the research samples and all training samples should be determined to locate the  $k$  nearest neighbors. It necessitates a significant amount of computing overhead in the case of big data. To discover the  $k$

Negative	TN	FN	Negative Predictive Value (NPV) FOR
Positive	FP	TP	Precision (PPV) FDR
	Specificity (TNR) FPR	Sensitivity (TPR) FNR	Accuracy Error
	Negative	Positive	

FIGURE 1: The confusion matrix.

nearest neighbors total training sets, some researchers use distributed frameworks like Hadoop [86]. These methods usually yield the same  $k$  nearest neighbors but at the expense of a massively distributed system. On the other hand, other authors consider searching for the closest neighbors in a smaller training data set. Using a KNN classifier on big data necessitates many computing resources. The class mark of a test sample is calculated using the  $k$  closest samples from the training data set in this classification process. The distance between the research samples and all training samples should be measured to locate the  $k$  closest neighbors [86].

**3.5. Linear Discriminant Analysis (LDA).** Fisher's linear discriminant is a statistical and another tool for evaluating a linear mixture of features that describes or distinguishes two or more types of objects or events. Fisher's linear discriminant is a generalization of LDA. The resulting combination may be utilized as a linear classifier or, more broadly, as a dimensionality reducer before additional classification. Discriminant analysis is employed where categories are known a priori (unlike in cluster analysis). Each scenario requires a score on one or more quantitative predictor variables and a score on a group indicator [87]. In its most abstract form, discriminant function analysis involves grouping, classifying, or categorizing objects into related groups, classes, or categories.

**3.6. Performance Metrics.** Patients are assigned to one of the four cells identified as  $d$  in Figure 1 according to classification outcomes and regardless of whether or not the target diagnosis is focused on the classification result and whether this evaluation has produced either a positive outcome (the individual seems to be the person) or a negative outcome (the person does not seem to have the condition) (the person seems not to have the condition). The numbers of individuals in each of the four cells will then be employed to calculate sensitivity, specificity, and predictive values, which are based on the following formulas [88] as expressed as percentages:

TABLE 2: Different parts of brain as feature of diagnosis.

1. Orbital frontal cortex	2. Anterior cingulate	3. Putamen
4. Prefrontal cortex	5. Posterior cingulate	6. Putamen LR
7. Superior frontal cortex	8. Occipital	9. Putamen L
10. Lateral temporal cortex	11. Global cortex	12. Putamen R
13. Medial temporal cortex	14. Amygdala	15. Putamen La
16. Posterior precuneus	17. Hippocampus	18. Putamen Lp
19. Ventral striatum	20. Caudate	21. Putamen Ra
22. Ventral striatum _LR	23. Caudate _LR	24. Putamen RP
25. Pons	26. Thalamus	27. Raphe
28. Gray matter VBM8	29. Substantia nigra	30. Raphe dorsal
31. White matter VBM8	32. Midbrain	33. Raphe nuclei
34. Brain mask GM_WM_CSF	35. Medulla	36. Centrum semiovale
37. Parietal		

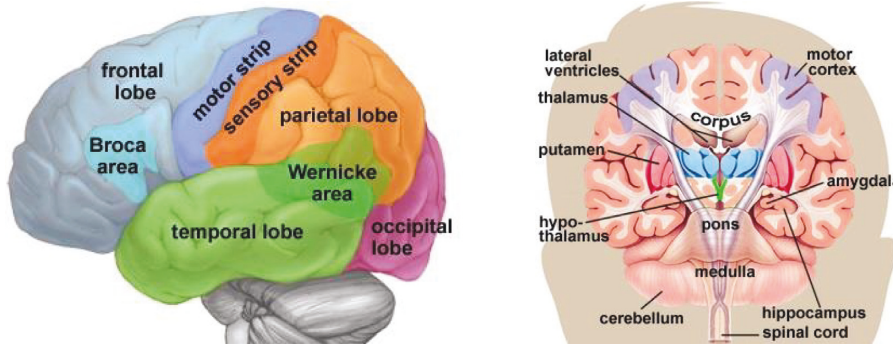


FIGURE 2: The anatomy of the cerebrum in the human brain [89].

$$\text{Sensitivity} = \left[ \frac{TP}{TP + FN} \right] \times 100,$$

$$\text{Specificity} = \left[ \frac{TN}{TN + FP} \right] \times 100,$$

$$\text{Positive predictive value (PPV)} = \left[ \frac{TP}{TP + FP} \right] \times 100,$$

$$\text{Negative predictive value (NPV)} = \left[ \frac{TN}{TN + FN} \right] \times 100,$$

$$\text{Accuracy (ACC)} = \left[ \frac{TP + TN}{TP + TN + FP + FN} \right] \times 100.$$

(1)

These are the criteria cited by researchers and clinicians related to sensitivity, specificity, and predictive values to determine the impact of a classifier outcome—i.e., often as percentages but usually as decimal fractions, preferably with an acceptable confidence interval of 95 percent. The simplicity, and even familiarity, of these four metrics, on the other hand, can obscure the existence of several complications that are often ignored. There could be flaws in either the comparison standard or the exam or both. The four metrics cannot be considered indisputable and unchangeable test characteristics: the rigor of the evaluation and the occurrence of the target condition in the study determine the measurements inserted into the cells of Figure 1 [88].

## 4. Results and Discussion

**4.1. Data Collection.** For this paper, the information was collected for the ADNI data set. The ADNI was founded by Principal Investigator Michael W. Weiner, MD, in 2003 as a public-private study. The primary objective of ADNI was to determine whether it was possible to track the progression of MCI and early AD with serial RMI, PET, and other biological markers, as well as clinical and neuropsychological evaluations. Several of the participants were able to obtain baseline and follow-up measurements of FDG. During the study trials, PET scans with 18F-AV45 as well as 11C-PiB were conducted for imaging of amyloid plaques. For each baseline and study, structural MRIs (1.5T or 3T, magnetization prepared rapid acquisition gradient echo) are obtained. The ADNI database also included Apolipoprotein E (APOE) genotypes, CSF scales, and clinical evaluations.

**4.2. Descriptive Statistics.** In this study, 75 topics were selected from the ADNI GWAS data set [89] with more than seven years of FDG PET, structural MRI, [18F] AV45, and [11C] PIB scans. All PET images accompanied by structural MRIs were imported into the ADNI database. The regions of interest (ROI) in a high-resolution MRI prototype were drawn manually. The variables used are presented in Table 2. The orbital cortical, prefrontal, superior frontal, lateral temporal, parietal, medial precuneus, occipital, anterior cingulate, and posterior cingulate make up the global cortex. The ROI of gray matter in the cerebellum is utilized as target

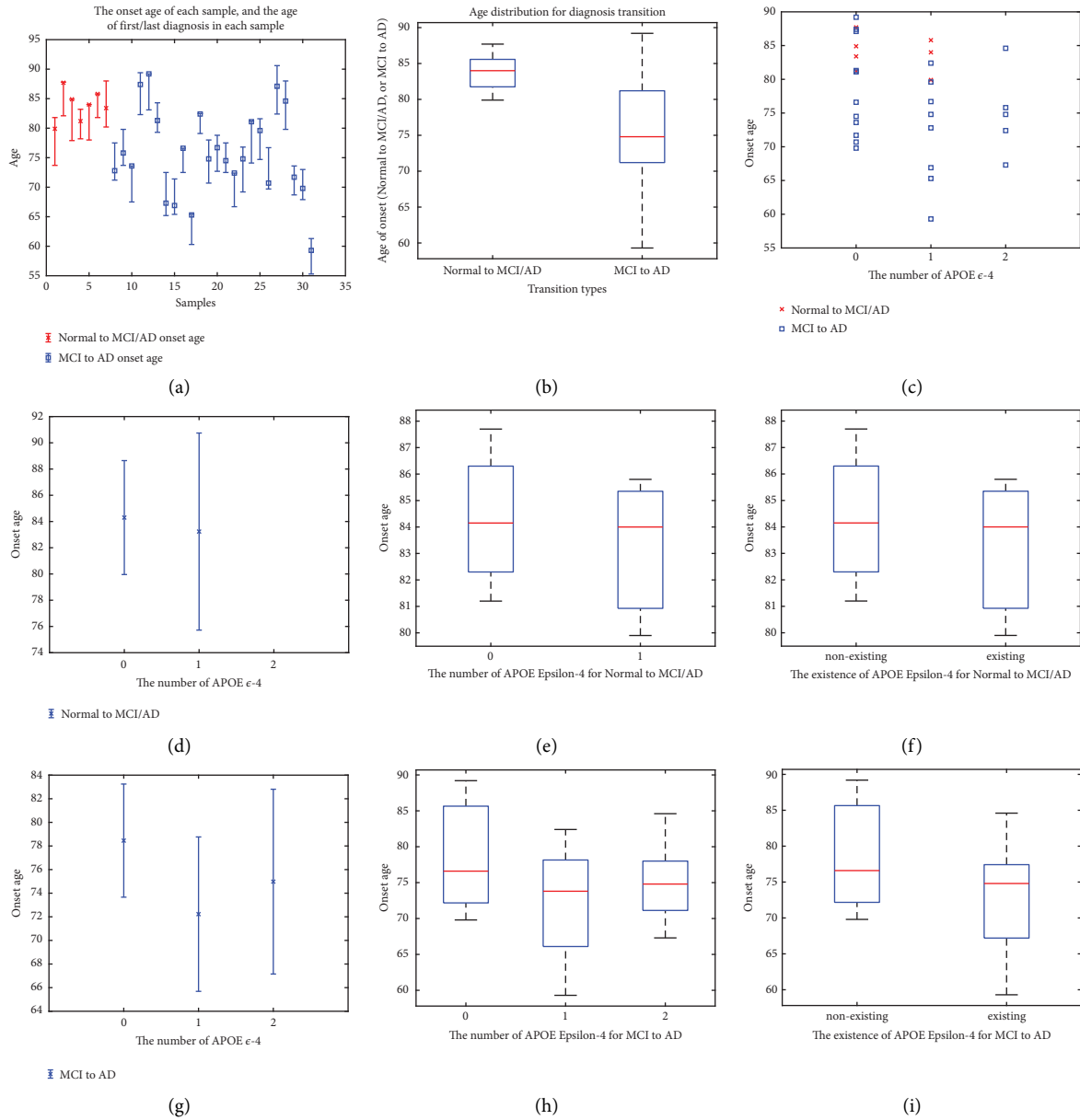


FIGURE 3: Results of descriptive statistics.

tissue, and the 34 ROIs in the normal MNI space that consists of cerebellum were used as template ROIs for all subjects. Refer to Figure 2 for an illustration of the brain's configuration.

The comparative study of the patients with Alzheimer's disease is seen in Figure 3. Longitudinal Alzheimer's research is critical because the abnormality and order of shifts with each biomarker vary dramatically as Alzheimer's progresses over time (see Figures 3(a) and 3(b)). The quantitative PET approach is regarded as a crucial method for tracking and assessing Alzheimer's disease development. Standardization and optimum use of PET in AD imaging include evaluating single or multiparametric PET output in the evaluation of patients. Based on the analysis of 32 patients, some of them changed the progression and Alzheimer's between normal

to MCI of Alzheimer and MCI to AD. Based on the results in Figure 3(b), people with new symptoms of AD and MCI is detected in almost 85 years old. In other words, this group saw the first effects of AD on their brains (age between [80 and 86]). However, the progression from MCI to AD is revealed for a wide range of ages [60–90] years with a mean of 75.

Regarding Figure 3, in the normal group, there was no discrepancy among converters and nonconverters in age, APOE carriers. It also exists in APOE- $\epsilon-4$  for decreasing the onset age of AD. That is why the age range of people with APOE- $\epsilon-4$  is lower than people without this carrier (see Figure 3(f)). It has occurred for people's AD progression. The progression of MCI to AD for people without APOE- $\epsilon-4$  has occurred for people for age between [70 and 90] (see Figure 4).

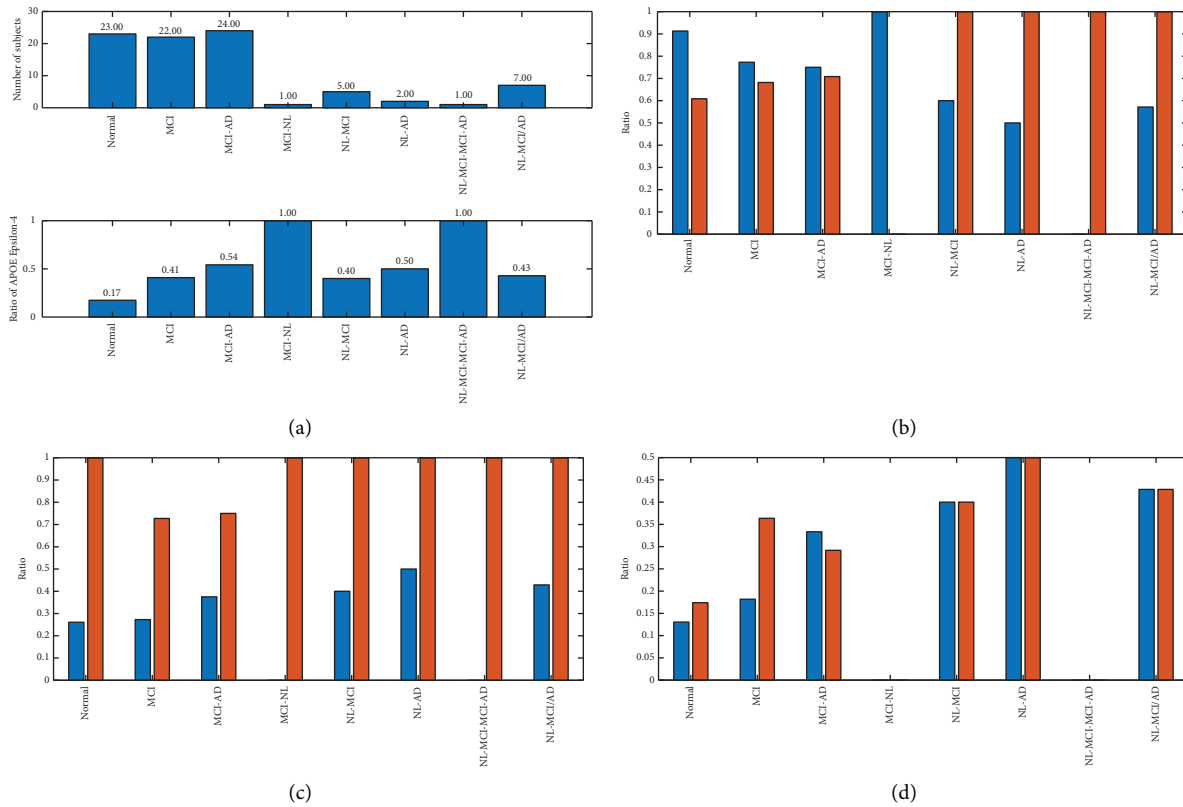


FIGURE 4: Results of diagnosis using statistical analysis.

TABLE 3: The SNP sequence involved.

SNP name	Sequence
rs1876152	CCGAGGTGACCTCAGGGAGGAACCGAGAAAGAAATACCCTGACTTCACCTC
rs1501228	ATTAGGTAGTCAGTTCTGCACAGAAGATATGCTTCTCGTCCAAATAAATG
rs1946867	CTTCATCTTTTTGTGTGGCAACATATGAAGCTGTACCAAATTGTATGTT

**4.3. Results of Diagnosis Using Statistical Analysis.** In this paper, three SNPs are used for the diagnosis of AD in the sample patients. The essential SNP is shown in Table 3.

The rs1876152 SNP has three variations in the sample domain: GG, GA, and AA. Three participants were selected to display their standard uptake values ratios in cortexes such as the pos-cingulate and pos-precuneus, frontal, parietal, and occipital. The presented findings support the hypothesis seen in Figure 5 that the detected SNP can substantially affect the decreasing pace of FDG uptake. The ApoE genotypes of the participants are all the same, which is  $\epsilon$ 4 and  $\epsilon$ 3. As a result, the various declining speeds are unrelated to the ApoE difference in this situation. According to the findings, the suggested SNPs have a more significant association with QTs than the SNP from the ApoE gene. The genotypes rs1876152 on chromosome 5, rs1501228 on chromosome 1, and rs1946867 on chromosome 4 all have a strong linear association with FDG, [18F]AV45, and [11C]PIB measurements, respectively. FDG, [18F]AV45, and [11C]PIB PET measurements all show a strong association with the genotypes rs1876152, rs1501228, and rs1946867, respectively. The ApoE genotype is a coarser genetic risk

factor for AD. To better track the progression of AD, our research identified genes that have strong associations with quantitative characteristics of three PET tracers other than the ApoE genotype. The current ADNI research will observe the assessment of the three genotypes in controlling AD development.

The Y-axis in Figure 5 indicates the average discrepancy in FDG measurements before and after the diagnosis process for seven years. The error bar represents the 95% confidence interval for the discrepancy in means. On the X-axis, the SNP genotype rs1876152 has three variations: GG, GA, and AA. The SNP genotype rs1501228 has three variants: GG, TG, and TT, while rs1946867 has three variants: GG, GA, and AA. The subjects with GG alleles have a minor difference in FDG measurements between two transformations (see Figure 5(a)). After the transition, the FDG SUVR decreases the most when the gap in AA alleles is more significant.

**4.4. Results of Diagnosis Using Machine Learning.** In this paper, 37 features are used for the diagnosis of AD in patients. The input features are indicated in Table 2 that are

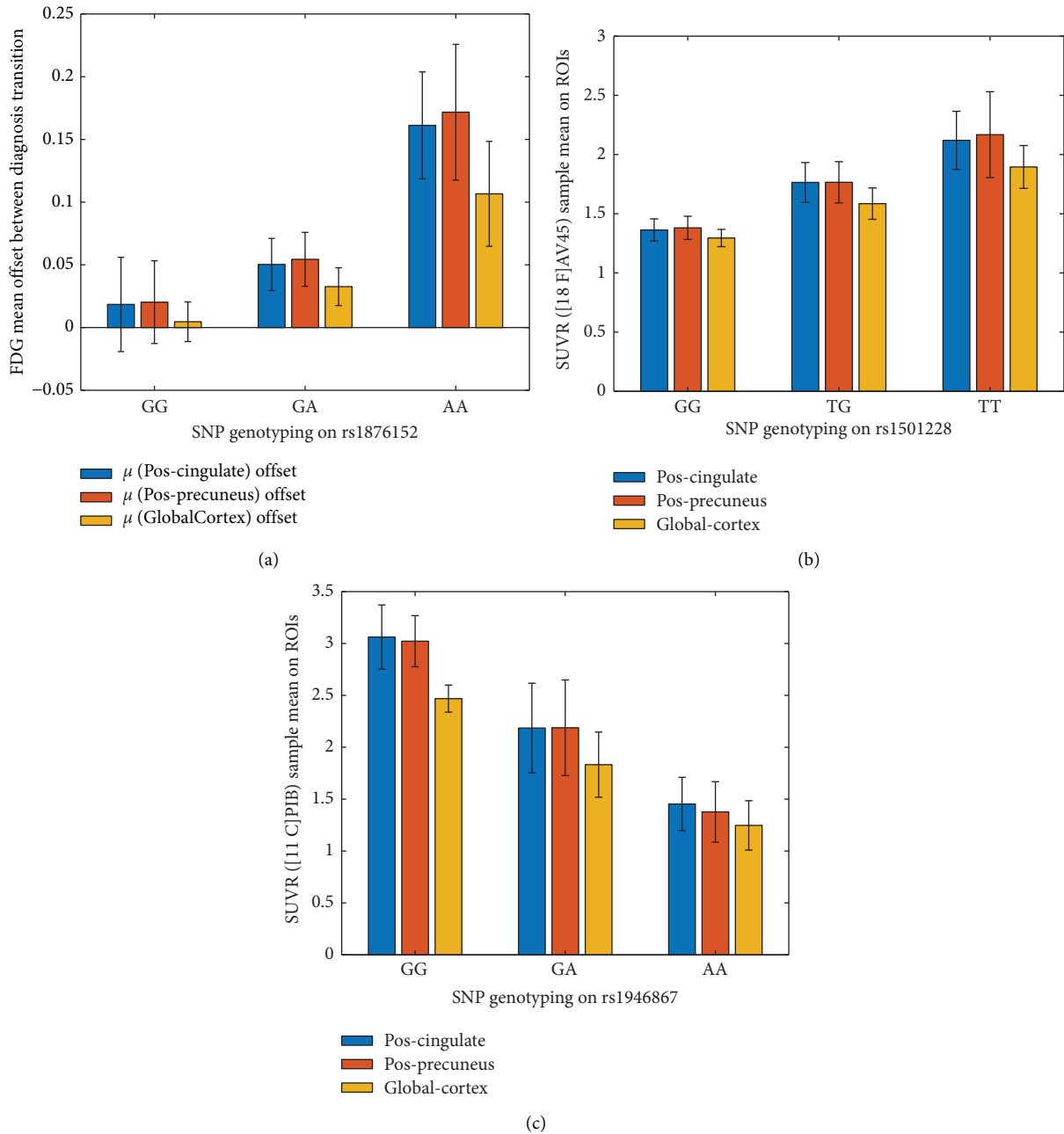


FIGURE 5: Standard uptake values ratios for three essential SNP.

different parts of the brain PET frequency based on PET images. For the implementation of machine learning methods, the main features should be normalized. The normalizations have been done in each variable to range data between  $-1$  and  $1$ . The next step is to decrease the number of variables. In this part, for reducing the feature principle component analysis of use. The normalized cumulative summation of sorted eigenvalues (NCSE) is illustrated in Figure 6. Based on the results of feature reduction, the first ten features have a 99% power of all 37 inputs.

In this paper, four machine learning methods consist of KNN, SVM, LDA, and CNN to diagnose AD. Ten reduced

features are used as input layers, and two labels of MCI and normal value are used as output layers. The results of classification are reported for 511 patients and 311 normal people. The performance metrics are illustrated as confusion matrix and ROC curve. For CNN methods, the training process is shown in Figure 7.

Pseudo-code of the presented procedure.

```

Collecting PET ()
For All (images)
    Feature extraction ()
End for A = Extract (Covariance Matrix)
    
```

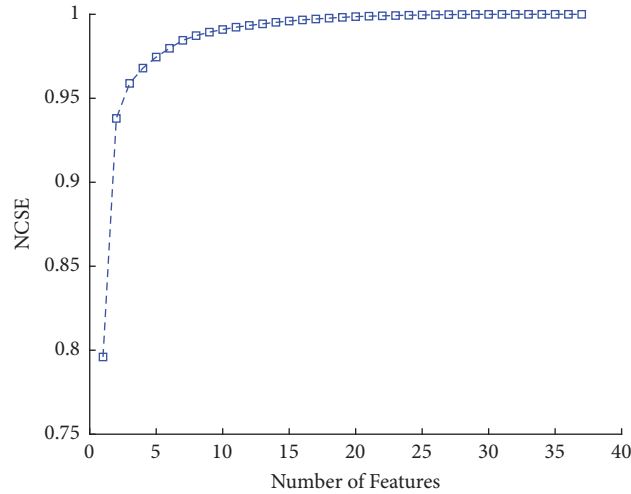


FIGURE 6: Normalized cumulative summation of sorted eigenvalues for feature reduction.

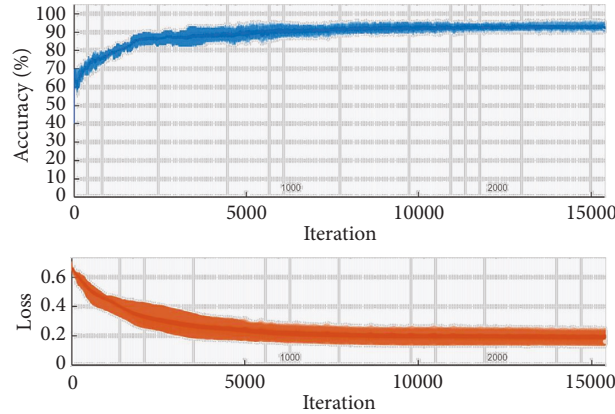


FIGURE 7: The training process of the CNN method.

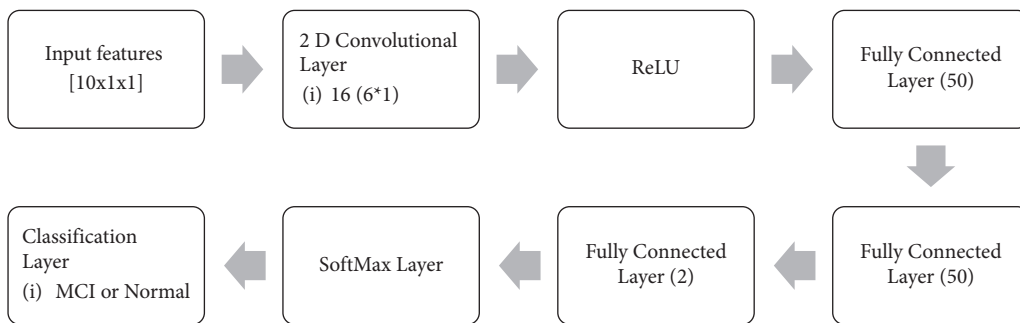


FIGURE 8: The architecture of the CNN method.

- Calculate (Eigen Vector (A))
- Feature reduction ()
- Classification ()
- Performance Analysis ()

Regarding Figure 7, the process is repeated until the accuracy and loss value are stable. Additionally, a convolutional layer and activation layer are employed for

classification, as shown in Figure 8. The ReLU activation function is the best choice for CNN techniques for removing negative values. Three fully connected layers with 50, 50, and 2 are used for changing data size to two categories. Finally, the SoftMax layer connects the architecture to the output layer. Results are indicated in the form of Figures 9 and 10.

Based on the confusion matrix, the green cells are true or correct in diagnosis versus orange cells as false diagnosis



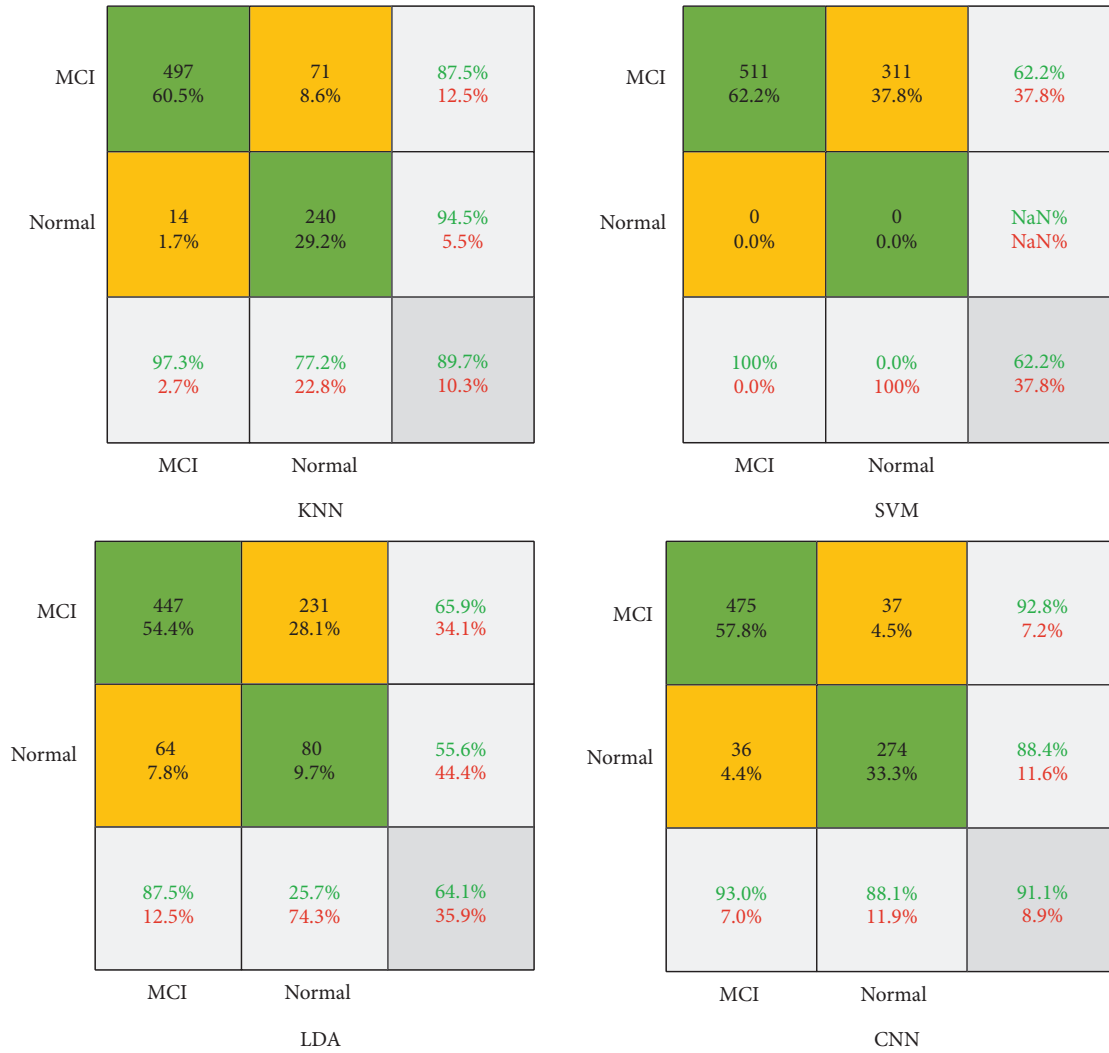


FIGURE 9: The confusion matrix of the presented classifiers.

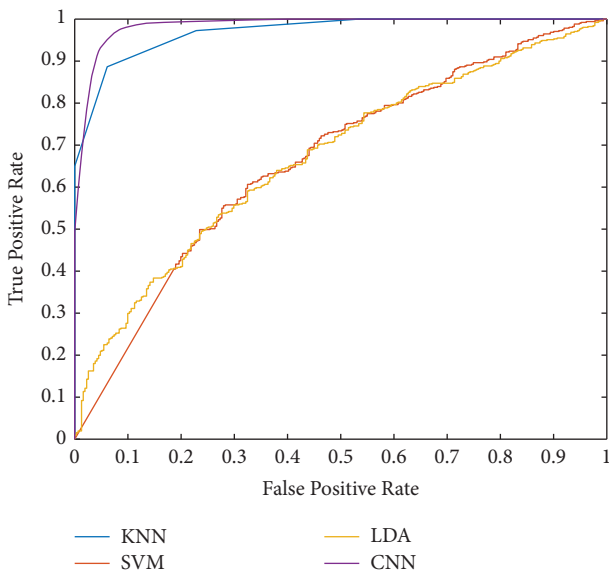


FIGURE 10: The ROC curve for the utilized machine learning classifiers.

value. Regarding the results of the KNN method in Figure 9, from 511 MCI samples, 497(97.3%) of them are detected successfully. In other words, the sensitivity of the KNN method is 97.3% for the diagnosis of MCI. On the other hand, from 311 normal samples, KNN finds 240(77.2%) of them correctly. This parameter is also called specificity. Based on the results of KNN, the precision of the method is 87.5%. It means that 497 + 71 persons are detected as MCI patients and that 87.5% of them are correct. Finally, the accuracy of the KNN is 89.7%. For a description of the SVM method, the method could not detect any types of patients and the accuracy are 62.2%. Nevertheless, the sensitivity of the SVM is 100%, and the specificity is zero. It means that none of the normal people is detected. About the LDA method, the sensitivity is 87.5%, while the specificity is 25.7%. It means that only 80(25.7%) normal persons are detected successfully. In this paper, we presented a CNN architecture to find an accurate model for AD diagnosis. Based on the results of the CNN method, the sensitivity is 93%. The CNN method could diagnose 88.1% of normal persons. Regarding the results of the classification, the

highest accuracy is belonging to CNN with 91.1%. The ROC curve is depicted in Figure 10 for a good description of the classifiers. Regarding Figure 10, the  $x$ -axis is the false-positive rate (FPR), and the  $y$ -axis is the true-positive rate (TPR). The method with lower FPR and higher TPR is desirable. Results show that the KNN and CNN method is a desirable method for diagnosing AD. However, the LDA and SVM are illustrated with lower accuracy.

## 5. Discussion

Clinically, image processing employing a CNN has gained considerable attention as a form of artificial intelligence. Its high performance in image recognition makes CNN a branch of deep neural networks (so-called deep learning) that is recognized to be highly useful for image analysis. A recent study employed a CNN to automatically diagnose tuberculosis from chest radiographs. Through the use of a CNN, we were also able to segment brain tumors and predict genotype from magnetic resonance images. One study found that dynamic contrast agent-enhanced computed tomography was very effective in distinguishing liver masses. PET/CT imaging has also been successfully used with CNN algorithms. In more recent years, generative adversarial networks (GANs) have been used to increase super-resolution efficiency, yet these approaches have been limited by the difficulty of training GANs, which is notoriously difficult. While deep neural networks have been effectively used for PET image denoising and radiation dose reduction in a number of recent articles, the application of deep learning for PET imaging is a less-explored research domain. The super-resolution issue, unlike the denoising problem, tries to build a clearer image from a hazy one while preferably maintaining edges. As a consequence, super-resolution requires different network architectures and data preparation procedures than denoising. The future work on super-resolution PET will utilize a diverse range of techniques, including both (classical) penalized deconvolution using joint entropy and deep learning using CNN. For future work, it is better to use some powerful feature extraction methods to select the more reliable features for diagnosis with PET images.

## 6. Conclusion

We examined genes that were significantly correlated with statistical properties of three PET tracers that are not associated with ApoE genotype for the purpose of monitoring AD progression. This article discusses 37 characteristics relevant to the diagnosis of Alzheimer's disease. A PET image provides inputs for different parts of the brain depending on their frequency dependence. We discovered several SNPs that have a link to PET imaging quantitative traits (QTs). Moreover, the classification is done to diagnose AD based on the frequency of different brain parts in PET images. The results are illustrated with performance analysis metrics. According to a study of patients, some improved their Alzheimer's development from mild to MCI and MCI to AD. According to the findings, individuals as young as 85 years old have additional signs of

Alzheimer's disease and MCI. In other words, this population saw the earliest signs of Alzheimer's disease in their brains (age between [80 and 86]). The progression from MCI to AD, on the other hand, is visible for a wide variety of ages [60–90] years, with a mean of 75. The genotypes rs1876152, rs1501228, and rs1946867, respectively, have a clear linear relationship with FDG, [18F] AV45, and [11C] PIB scales, according to the findings. According to the results, the proposed SNPs have a stronger connection to QTs than the SNP from the ApoE gene. Our study examined genes that have significant correlations with statistical properties of three PET tracers other than the ApoE genotype in order to help monitor the evolution of AD. In this article, 37 characteristics are used to diagnose Alzheimer's disease in patients. Different areas of the brain frequency dependent on PET images are used as input functions. According to the effects of feature reduction, the first ten functions have a 99 percent impact on all 37 inputs. KNN, SVM, LDA, and CNN are four machine learning approaches used to diagnose Alzheimer's disease. The input layer consists of ten reduced features, while the output layer consists of two MCI labels and the normal value. According to the findings of the KNN process, 497 (or 97.3 percent) of the 511 MCI samples were successfully detected. In other words, the KNN system has a sensitivity of 97.3 percent for diagnosing MCI. KNN, on the other hand, accurately identifies 240 (or 77.2 percent) of 311 standard samples. The SVM system failed to detect any of the patients, with a 62.2 percent accuracy. Despite this, the SVM's sensitivity is 100 percent, and its specificity is nil. CNN has the best accuracy rate of 91.1 percent when it comes to classification data. The findings suggest that the KNN and CNN methods are suitable for diagnosing Alzheimer's disease. The LDA and SVM, on the other hand, are depicted with less precision.

## Data Availability

For this report, the information was collected from the ADNI data set (<http://adni.loni.usc.edu/>).

## Conflicts of Interest

The authors declare that they have no conflicts of interest to report regarding the publication of this paper.

## References

- [1] S. Alladi, J. Xuereb, T. Bak et al., "Focal cortical presentations of Alzheimer's disease," *Brain*, vol. 130, no. 10, pp. 2636–2645, 2007.
- [2] M. Amini, M. M. Pedram, A. Moradi, and M. Ochani, "Plasmonics optoelectronics nanobiosensors for detection of Alzheimer's disease biomarker via amyloid-beta ( $\alpha\beta$ ) in near-infrared," *Plasmonics*, vol. 17, no. 3, pp. 1191–1201, 2022.
- [3] M. E. Murray, N. R. Graff-Radford, O. A. Ross, R. C. Petersen, R. Duara, and D. W. Dickson, "Neuropathologically defined subtypes of Alzheimer's disease with distinct clinical

- characteristics: a retrospective study,” *The Lancet Neurology*, vol. 10, no. 9, pp. 785–796, 2011.
- [4] G. M. McKhann, D. S. Knopman, H. Chertkow et al., “The diagnosis of dementia due to Alzheimer’s disease: recommendations from the National Institute on Aging–Alzheimer’s Association workgroups on diagnostic guidelines for Alzheimer’s disease,” *Alzheimer’s and Dementia*, vol. 7, no. 3, pp. 263–269, 2011.
- [5] B. Dubois, H. H. Feldman, C. Jacova et al., “Advancing research diagnostic criteria for Alzheimer’s disease: the IWG-2 criteria,” *The Lancet Neurology*, vol. 13, no. 6, pp. 614–629, 2014.
- [6] G. Bera, R. Migliaccio, T. Michelin et al., “Parietal involvement in the semantic variant of primary progressive aphasia with Alzheimer’s disease cerebrospinal fluid profile,” *Journal of Alzheimer’s Disease*, vol. 66, no. 1, pp. 271–280, 2018.
- [7] S. E. Lee, G. D. Rabinovici, M. C. Mayo et al., “Clinicopathological correlations in corticobasal degeneration,” *Annals of Neurology*, vol. 70, no. 2, pp. 327–340, 2011.
- [8] J. D. Warren, P. D. Fletcher, and H. L. Golden, “The paradox of syndromic diversity in Alzheimer disease,” *Nature Reviews Neurology*, vol. 8, no. 8, pp. 451–464, 2012, <http://www.nature.com/doi/10.1038/nrneuro.2012.135>.
- [9] S. A. Mofrad, A. J. Lundervold, A. Vik, and A. S. Lundervold, “Cognitive and MRI trajectories for prediction of Alzheimer’s disease,” *Scientific Reports*, vol. 11, no. 1, 2021.
- [10] J. Giorgio, S. Landau, W. J. Jagust, and Z. Kourtzi, “Modelling prognostic trajectories of cognitive decline due to Alzheimer’s disease,” *NeuroImage: Clinical*, vol. 26, 2020.
- [11] L. Mosconi, W. H. Tsui, K. Herholz et al., “Multicenter standardized 18F-FDG PET diagnosis of mild cognitive impairment, Alzheimer’s disease, and other dementias,” *Journal of Nuclear Medicine*, vol. 49, no. 3, pp. 390–398, 2008.
- [12] L. C. de Souza, F. Corlier, M.-O. Habert et al., “Similar amyloid- $\beta$  burden in posterior cortical atrophy and Alzheimer’s disease,” *Brain*, vol. 134, no. 7, pp. 2036–2043, 2011.
- [13] X. Zhao, X. Zhang, Z. Cai et al., “Chaos enhanced grey wolf optimization wrapped ELM for diagnosis of paraquat-poisoned patients,” *Computational Biology and Chemistry*, vol. 78, pp. 481–490, 2019.
- [14] R. Ossenkoppele, D. R. Schonhaut, M. Schöll et al., “Tau PET patterns mirror clinical and neuroanatomical variability in Alzheimer’s disease,” *Brain*, vol. 139, no. 5, pp. 1551–1567, 2016.
- [15] S. L. Risacher, L. Shen, J. D. West et al., “Longitudinal MRI atrophy biomarkers: relationship to conversion in the ADNI cohort,” *Neurobiology of Aging*, vol. 31, no. 8, pp. 1401–1418, 2010.
- [16] L. Shen, S. Kim, S. L. Risacher et al., “Whole genome association study of brain-wide imaging phenotypes for identifying quantitative trait loci in MCI and AD: a study of the ADNI cohort,” *NeuroImage*, vol. 53, no. 3, pp. 1051–1063, 2010.
- [17] X. Yao, J. Yan, S. Kim et al., “Two-dimensional enrichment analysis for mining high-level imaging genetic associations,” *Brain Informatics*, vol. 4, no. 1, pp. 27–37, 2017.
- [18] D. P. Hibar, J. L. Stein, O. Kohannim et al., “Voxelwise genome-wide association study (vGeneWas): multivariate gene-based association testing in 731 elderly subjects,” *NeuroImage*, vol. 56, no. 4, pp. 1875–1891, 2011.
- [19] He Shida, F. Guo, Q. Zou, and H. Ding, “MRMD2. 0: a python tool for machine learning with feature ranking and reduction,” *Current Bioinformatics*, vol. 15, no. 10, pp. 1213–1221, 2021.
- [20] C. Juntao, Q. Zou, and J. Li, “DeepM6ASeq-EL: prediction of human N6-methyladenosine (m6A) sites with LSTM and ensemble learning,” *Frontiers of Computer Science*, vol. 16, no. 2, pp. 162302–7, 2022.
- [21] Q. Xu, Q. Guo, C.-X. Wang et al., “Network differentiation: a computational method of pathogenesis diagnosis in traditional Chinese medicine based on systems science,” *Artificial Intelligence in Medicine*, vol. 118, Article ID 102134, 2021.
- [22] M. Maruyama, H. Shimada, T. Suhara et al., “Imaging of tau pathology in a tauopathy mouse model and in Alzheimer patients compared to normal controls,” *Neuron*, vol. 79, no. 6, pp. 1094–1108, 2013.
- [23] H. Shimada, S. Kitamura, H. Shinotoh et al., “Association between A $\beta$  and tau accumulations and their influence on clinical features in aging and Alzheimer’s disease spectrum brains: a [11C] PBB3-PET study,” *Alzheimer’s and Dementia: Diagnosis, Assessment & Disease Monitoring*, vol. 6, no. 1, pp. 11–20, 2017.
- [24] K. Chiotis, P. Stenkrona, O. Almkvist et al., “Dual tracer tau PET imaging reveals different molecular targets for 11 C-THK5351 and 11 C-PBB3 in the Alzheimer brain,” *European Journal of Nuclear Medicine and Molecular Imaging*, vol. 45, no. 9, pp. 1605–1617, 2018.
- [25] K. Chiotis, L. Saint-Aubert, I. Savitcheva et al., “Imaging in vivo tau pathology in Alzheimer’s disease with THK5317 PET in a multi-modal paradigm,” *European Journal of Nuclear Medicine and Molecular Imaging*, vol. 43, no. 9, pp. 1686–1699, 2016.
- [26] S. N. Lockhart, S. L. Baker, N. Okamura et al., “Dynamic PET measures of tau accumulation in cognitively normal older adults and Alzheimer’s disease patients measured using [18F] THK-5351,” *PLoS One*, vol. 11, no. 6, Article ID e0158460, 2016.
- [27] L. Saint-Aubert, O. Almkvist, K. Chiotis, R. Almeida, A. Wall, and A. Nordberg, “Regional tau deposition measured by [18F] THK5317 positron emission tomography is associated to cognition via glucose metabolism in Alzheimer’s disease,” *Alzheimer’s Research & Therapy*, vol. 8, no. 1, 2016.
- [28] J. Chen, Y. Li, E. Pirraglia, N. Okamura, H. Rusinek, and M. J. de Leon, “Quantitative evaluation of tau PET tracers 18F-THK5351 and 18F-AV-1451 in Alzheimer’s disease with standardized uptake value peak-alignment (SUVP) normalization,” *European Journal of Nuclear Medicine and Molecular Imaging*, vol. 45, no. 9, pp. 1596–1604, 2018.
- [29] L. Passamonti, P. Vázquez Rodríguez, Y. T. Hong et al., “18F-AV-1451 positron emission tomography in Alzheimer’s disease and progressive supranuclear palsy,” *Brain*, vol. 140, no. 3, Article ID aww340, 2017.
- [30] L. Wang, T. L. Benzinger, Y. Su et al., “Evaluation of tau imaging in staging Alzheimer disease and revealing interactions between  $\beta$ -amyloid and tauopathy,” *JAMA Neurology*, vol. 73, no. 9, 2016.
- [31] H. Cho, J. Y. Choi, M. S. Hwang et al., “Tau PET in Alzheimer disease and mild cognitive impairment,” *Neurology*, vol. 87, no. 4, pp. 375–383, 2016.
- [32] K. A. Johnson, A. Schultz, R. A. Betensky et al., “Tau positron emission tomographic imaging in aging and early Alzheimer disease,” *Annals of Neurology*, vol. 79, no. 1, pp. 110–119, 2016.
- [33] M. J. Pontecorvo, M. D. Devous, M. Navitsky et al., “Relationships between flortaucipir PET tau binding and amyloid burden, clinical diagnosis, age and cognition,” *Brain*, vol. 140, no. 3, pp. 748–763, 2017.
- [34] J. L. Whitwell, J. Graff-Radford, N. Tosakulwong et al., “[18F] AV-1451 clustering of entorhinal and cortical uptake in

- Alzheimer's disease," *Annals of Neurology*, vol. 83, no. 2, pp. 248–257, 2018.
- [35] N. A. Murugan, K. Chiotis, E. Rodriguez-Vieitez, L. Lemoine, H. Ågren, and A. Nordberg, "Cross-interaction of tau PET tracers with monoamine oxidase B: evidence from in silico modelling and in vivo imaging," *European Journal of Nuclear Medicine and Molecular Imaging*, vol. 46, no. 6, pp. 1369–1382, 2019.
- [36] T. G. Lohith, I. Bennacef, R. Vandenberghe et al., "Brain imaging of Alzheimer dementia patients and elderly controls with 18F-MK-6240, a PET tracer targeting neurofibrillary tangles," *Journal of Nuclear Medicine*, vol. 60, no. 1, pp. 107–114, 2019.
- [37] H. Kroth, F. Oden, J. Molette et al., "Discovery and pre-clinical characterization of [18 F] PI-2620, a next-generation tau PET tracer for the assessment of tau pathology in Alzheimer's disease and other tauopathies," *European Journal of Nuclear Medicine and Molecular Imaging*, vol. 46, no. 10, pp. 2178–2189, 2019.
- [38] D. F. Wong, R. A. Comley, H. Kuwabara et al., "Characterization of 3 novel tau radiopharmaceuticals, 11C-RO-963, 11C-RO-643, and 18F-RO-948, in healthy controls and in Alzheimer subjects," *Journal of Nuclear Medicine*, vol. 59, no. 12, pp. 1869–1876, 2018.
- [39] S. Sanabria Bohórquez, J. Marik, A. Ogasawara et al., "[18F] GTP1 (Genentech tau probe 1), a radioligand for detecting neurofibrillary tangle tau pathology in Alzheimer's disease," *European Journal of Nuclear Medicine and Molecular Imaging*, vol. 46, no. 10, pp. 2077–2089, 2019.
- [40] L. Du, K. Liu, X. Yao et al., "Detecting genetic associations with brain imaging phenotypes in Alzheimer's disease via a novel structured SCCA approach," *Medical Image Analysis*, vol. 61, Article ID 101656, 2020.
- [41] X. Yao, S. Cong, J. Yan et al., "Regional imaging genetic enrichment analysis," *Bioinformatics*, vol. 36, no. 8, pp. 2554–2560, 2020.
- [42] Yi Zou, H. Wu, X. Guo et al., "MK-FSVM-SVDD: a multiple kernel-based fuzzy SVM model for predicting DNA-binding proteins via support vector data description," *Current Bioinformatics*, vol. 16, no. 2, pp. 274–283, 2021.
- [43] Y. Yan, L. Feng, M. Shi, C. Cui, and Y. Liu, "Effect of plasma-activated water on the structure and in vitro digestibility of waxy and normal maize starches during heat-moisture treatment," *Food chemistry*, vol. 306, Article ID 125589, 2020.
- [44] M. Shi, F. Wang, P. Lan et al., "Effect of ultrasonic intensity on structure and properties of wheat starch-monoacylglyceride complex and its influence on quality of norther-style Chinese steamed bread," *Lwt*, vol. 138, Article ID 110677, 2021.
- [45] N. Nejatishahidin, P. Fayyazanavi, and K. Jana, "Object pose estimation using mid-level visual representations," 2022, <https://arxiv.org/pdf/2203.01449.pdf>.
- [46] E. Eslami and H. -B. Yun, "Attention-based multi-scale convolutional neural network (A+ MCNN) for multi-class classification in road images," *Sensors*, vol. 21, no. 15, p. 5137, 2021.
- [47] T. Dubois, C. Reynaert, D. Jacques, B. Lepiece, and N. Zdanowicz, "From family surroundings to intestinal flora, A literature review concerning epigenetic processes in psychiatric disorders," *Psychiatria Danubina*, vol. 32, no. Suppl 1, pp. 158–163, 2020.
- [48] B. Wang, "Early warning method of marine products network marketing risk based on BP neural network algorithm," *Journal of Coastal Research*, vol. 103, no. sp1, pp. 177–181, 2020.
- [49] R. Prasad and K. D. Yadav, "Use of response surface methodology and artificial neural network approach for methylene blue removal by adsorption onto water hyacinth," *Water Conservation and Management*, vol. 4, no. 2, pp. 83–89, 2020.
- [50] M. Rezaei, F. Farahanipad, A. Dillhoff, R. Elmasri, and V. Athitsos, "Weakly-supervised hand part segmentation from depth images," in *Proceedings of the 14th Pervasive Technologies Related to Assistive Environments Conference*, pp. 218–225, Corfu Greece, June 2021.
- [51] A. Chandra, P. E. Valkimadi, G. Pagano, O. Cousins, G. Dervenoulas, and M. Politis, "Applications of amyloid, tau, and neuroinflammation PET imaging to Alzheimer's disease and mild cognitive impairment," *Human Brain Mapping*, vol. 40, no. 18, pp. 5424–5442, 2019.
- [52] H. Chen, A. A. Heidari, H. Chen, M. Wang, Z. Pan, and A. H. Gandomi, "Multi-population differential evolution-assisted Harris hawks optimization: framework and case studies," *Future Generation Computer Systems*, vol. 111, pp. 175–198, 2020.
- [53] X. Zhao, D. Li, B. Yang, C. Ma, Y. Zhu, and H. Chen, "Feature selection based on improved ant colony optimization for online detection of foreign fiber in cotton," *Applied Soft Computing*, vol. 24, pp. 585–596, 2014.
- [54] Y. Xu, H. Chen, J. Luo, Q. Zhang, S. Jiao, and X. Zhang, "Enhanced Moth-flame optimizer with mutation strategy for global optimization," *Information Sciences*, vol. 492, pp. 181–203, 2019.
- [55] C. Li, L. Hou, B. Y. Sharma et al., "Developing a new intelligent system for the diagnosis of tuberculous pleural effusion," *Computer Methods and Programs in Biomedicine*, vol. 153, pp. 211–225, 2018.
- [56] J. Xia, H. Chen, Q. Li et al., "Ultrasound-based differentiation of malignant and benign thyroid Nodules: an extreme learning machine approach," *Computer Methods and Programs in Biomedicine*, vol. 147, pp. 37–49, 2017.
- [57] H.-L. Chen, G. Wang, C. Ma, Z.-N. Cai, W.-B. Liu, and S.-J. Wang, "An efficient hybrid kernel extreme learning machine approach for early diagnosis of Parkinson's disease," *Neurocomputing*, vol. 184, pp. 131–144, 2016.
- [58] L. Hu, G. Hong, J. Ma, X. Wang, and H. Chen, "An efficient machine learning approach for diagnosis of paraquat-poisoned patients," *Computers in Biology and Medicine*, vol. 59, pp. 116–124, 2015.
- [59] X. Xu and H.-L. Chen, "Adaptive computational chemotaxis based on field in bacterial foraging optimization," *Soft Computing*, vol. 18, no. 4, pp. 797–807, 2014.
- [60] D. Zhao, L. Liu, F. Yu, A. A. Heidari, M. Wang, and G. Liang, "Chaotic random spare ant colony optimization for multi-threshold image segmentation of 2D Kapur entropy," *Knowledge-Based Systems*, Article ID 106510, 2020.
- [61] C. Yu, M. Chen, K. Cheng et al., "SGOA: annealing-behaved grasshopper optimizer for global tasks," *Engineering with Computers*, 2021.
- [62] H. Yu, W. Li, C. Chen et al., "Dynamic Gaussian bare-bones fruit fly optimizers with abandonment mechanism: method and analysis," *Engineering with Computers*, 2020.
- [63] M. Amini and M. M. Pedram, "Application of machine learning methods in diagnosis of alzheimer disease based on fractal feature extraction and convolutional neural network," in *Proceedings of the 9th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS)*, pp. 1–5, Bam, Iran, March 2022.
- [64] J. Tu, C. Huiling, and L. Jiacong, "Evolutionary biogeography-based Whale optimization methods with communication

- structure: towards measuring the balance,” *Knowledge-Based Systems*, Article ID 106642, 2020.
- [65] M. Amini, M. M. Pedram, A. R. Moradi, and M. Ouchani, “Diagnosis of alzheimer’s disease by time-dependent power spectrum descriptors and convolutional neural network using EEG signal,” *Computational and Mathematical Methods in Medicine*, vol. 2021, Article ID 5511922, 17 pages, 2021.
- [66] Y. Zhang, R. Liu, X. Wang, H. Chen, and C. Li, “Boosted binary Harris hawks optimizer and feature selection,” *Engineering with Computers*, vol. 37, no. 4, pp. 3741–3770, 2020.
- [67] M. Wang, H. Chen, B. Yang et al., “Toward an optimal kernel extreme learning machine using a chaotic moth-flame optimization strategy with applications in medical diagnoses,” *Neurocomputing*, vol. 267, pp. 69–84, 2017.
- [68] S. Hassantabar, M. Ahmadi, and A. Sharifi, “Diagnosis and detection of infected tissue of COVID-19 patients based on lung X-ray image using convolutional neural network approaches,” *Chaos, Solitons & Fractals*, vol. 140, Article ID 110170, 2020.
- [69] M. Ahmadi and R. Taghizadeh, “A gene expression programming model for economy growth using knowledge-based economy indicators: a comparison of GEP model and ARDL bounds testing approach,” *Journal of Modelling in Management*, vol. 14, no. 1, pp. 31–48, 2019.
- [70] S. Dorosti, S. Jafarzadeh Ghoushchi, E. Sobhrakhshankhah, M. Ahmadi, and A. Sharifi, “Application of gene expression programming and sensitivity analyses in analyzing effective parameters in gastric cancer tumor size and location,” *Soft Computing*, vol. 24, no. 13, pp. 9943–9964, 2020.
- [71] M. Ahmadi, A. Sharifi, S. Hassantabar, and S. Enayati, “QAIS-DSNN: tumor area segmentation of MRI image with optimized quantum matched-filter technique and deep spiking neural network,” *BioMed Research International*, vol. 2021, pp. 2021–16, 2021.
- [72] M. Ahmadi, A. Sharifi, M. Jafarian Fard, and N. Soleimani, “Detection of brain lesion location in MRI images using convolutional neural network and robust PCA,” *International Journal of Neuroscience*, vol. 4, pp. 1–12, 2021.
- [73] A. Sharifi, M. Ahmadi, M. A. Mehni, S. Jafarzadeh Ghoushchi, and Y. Pourasad, “Experimental and numerical diagnosis of fatigue foot using convolutional neural network,” *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 24, no. 16, pp. 1828–1840, 2021.
- [74] J. Artin, A. Valizadeh, M. Ahmadi, S. A. P. Kumar, and A. Sharifi, “Presentation of a novel method for prediction of traffic with climate condition based on ensemble learning of neural architecture search (NAS) and linear regression,” *Complexity*, vol. 2021, 13 pages, 2021.
- [75] M. Ahmadi, A. Taghavarashidizadeh, D. Javaheri, A. Masoumian, S. J. Ghoushchi, and Y. Pourasad, “DQRE-SCnet: a novel hybrid approach for selecting users in federated learning with deep-Q-reinforcement learning based on spectral clustering,” *Journal of King Saud University-Computer and Information Sciences*, 2021.
- [76] E. M. Onyema, P. K. Shukla, S. Dalal, M. N. Mathur, M. Zakariah, and B. Tiwari, “Enhancement of patient facial recognition through deep learning algorithm: ConvNet,” *Journal of Healthcare Engineering*, vol. 2021, 8 pages, 2021.
- [77] S. A. Butt, M. W. Anjum, S. A. Hassan, A. Garai, and E. M. Onyema, “Smart health application for remote tracking of ambulatory patients,” *Smart Healthcare System Design: Security and Privacy Aspects*, vol. 17, pp. 33–55, 2022.
- [78] L. Abualigah, D. Yousri, M. Abd Elaziz, A. A. Ewees, M. A. Al-Qaness, and A. H. Gandomi, “Aquila optimizer: a novel meta-heuristic optimization algorithm,” *Computers & Industrial Engineering*, vol. 157, Article ID 107250, 2021.
- [79] L. Abualigah, M. A. Elaziz, P. Sumari, Z. W. Geem, and A. H. Gandomi, “Reptile Search Algorithm (RSA): a nature-inspired meta-heuristic optimizer,” *Expert Systems with Applications*, vol. 191, Article ID 116158, 2022.
- [80] L. Abualigah, A. Diabat, P. Sumari, and A. H. Gandomi, “Applications, deployments, and integration of internet of drones (iod): a review,” *IEEE Sensors Journal*, vol. 21, no. 22, Article ID 25532, 2021.
- [81] R. Blasberg, “PET imaging of gene expression,” *European Journal of Cancer*, vol. 38, no. 16, pp. 2137–2146, 2002.
- [82] M. Ouchani, S. Gharibzadeh, M. Jamshidi, and M. Amini, “A review of methods of diagnosis and complexity analysis of alzheimer’s disease using EEG signals,” *BioMed Research International*, vol. 2021, pp. 1–15, 2021.
- [83] Y. Zhang, R. Liu, A. A. Heidari, X. Wang, Y. Chen, and M. Wang, “Towards augmented kernel extreme learning models for bankruptcy prediction: algorithmic behavior and comprehensive analysis,” *Neurocomputing*, vol. 430, 2020.
- [84] M. Wang and H. Chen, “Chaotic multi-swarm whale optimizer boosted support vector machine for medical diagnosis,” *Applied Soft Computing*, vol. 88, Article ID 105946, 2020.
- [85] D. A. Pisner and D. M. Schnyer, “Chapter 6 - support vector machine,” *Machine Learning*, Academic Press, Massachusetts, MA, USA, 2020.
- [86] H. Saadatfar, S. Khosravi, J. H. Joloudari, A. Mosavi, and S. Shamsirband, “A new K-nearest neighbors classifier for big data based on efficient data pruning,” *Mathematics*, vol. 8, no. 2, 2020.
- [87] Ş. Büyüköztürk and Ö. Çokluk-Bökeoğlu, “Discriminant function analysis: concept and application,” *Eğitim araştırmaları dergisi*, no. 33, pp. 73–92, 2008.
- [88] R. Trevehan, “Sensitivity, specificity, and predictive values: foundations, plabilities, and pitfalls in research and practice,” *Frontiers in Public Health*, vol. 5, 2017.
- [89] A. J. Saykin, L. Shen, T. M. Foroud et al., “Alzheimer’s Disease Neuroimaging Initiative biomarkers as quantitative phenotypes: genetics core aims, progress, and plans,” *Alzheimer’s and Dementia*, vol. 6, no. 3, pp. 265–273, 2010.

## Research Article

# Weighted Joint Sentiment-Topic Model for Sentiment Analysis Compared to ALGA: Adaptive Lexicon Learning Using Genetic Algorithm

Amjad Osmani <sup>1,2</sup> and Jamshid Bagherzadeh Mohasefi<sup>3</sup>

<sup>1</sup>Department of Computer Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran

<sup>2</sup>Department of Computer Engineering, Urmia Branch, Islamic Azad University, Urmia, Iran

<sup>3</sup>Department of Computer Engineering, Urmia University, Urmia, Iran

Correspondence should be addressed to Amjad Osmani; a.osmani@qiau.ac.ir

Received 23 February 2022; Revised 14 April 2022; Accepted 8 May 2022; Published 31 July 2022

Academic Editor: Mohamed Abdelaziz

Copyright © 2022 Amjad Osmani and Jamshid Bagherzadeh Mohasefi. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Latent Dirichlet Allocation (LDA) is an approach to unsupervised learning that aims to investigate the semantics among words in a document as well as the influence of a subject on a word. As an LDA-based model, Joint Sentiment-Topic (JST) examines the impact of topics and emotions on words. The emotion parameter is insufficient, and additional parameters may play valuable roles in achieving better performance. In this study, two new topic models, Weighted Joint Sentiment-Topic (WJST) and Weighted Joint Sentiment-Topic 1 (WJST1), have been presented to extend and improve JST through two new parameters that can generate a sentiment dictionary. In the proposed methods, each word in a document affects its neighbors, and different words in the document may be affected simultaneously by several neighbor words. Therefore, proposed models consider the effect of words on each other, which, from our view, is an important factor and can increase the performance of baseline methods. Regarding evaluation results, the new parameters have an immense effect on model accuracy. While not requiring labeled data, the proposed methods are more accurate than discriminative models such as SVM and logistic regression in accordance with evaluation results. The proposed methods are simple with a low number of parameters. While providing a broad perception of connections between different words in documents of a single collection (single-domain) or multiple collections (multidomain), the proposed methods have prepared solutions for two different situations (single-domain and multidomain). WJST is suitable for multidomain datasets, and WJST1 is a version of WJST which is suitable for single-domain datasets. While being able to detect emotion at the level of the document, the proposed models improve the evaluation outcomes of the baseline approaches. Thirteen datasets with different sizes have been used in implementations. In this study, perplexity, opinion mining at the level of the document, and topic\_coherency are employed for assessment. Also, a statistical test called Friedman test is used to check whether the results of the proposed models are statistically different from the results of other algorithms. As can be seen from results, the accuracy of proposed methods is above 80% for most of the datasets. WJST1 achieves the highest accuracy on Movie dataset with 97 percent, and WJST achieves the highest accuracy on Electronic dataset with 86 percent. The proposed models obtain better results compared to Adaptive Lexicon learning using Genetic Algorithm (ALGA), which employs an evolutionary approach to make an emotion dictionary. Results show that the proposed methods perform better with different topic number settings, especially for WJST1 with 97% accuracy at  $|Z|=5$  on the Movie dataset.

## 1. Introduction

Opinion extraction is one of the main branches of natural language processing (NLP) research. Comment extraction (emotion analysis) now is widely used in websites containing

different types of merchandise. Online product reviews can help customers buy a product and help manufacturers discover new opportunities by analyzing user feedback. Consequently, automated analysis of reviews is critical. Emotion Analyzer can browse comments on the web and categorize many comments

as positive or negative tags. This research is important because it makes managing customer requests easier and more efficient because product owners automatically extract customer feedback and use customer feedback to sell products. There are different methods for extracting opinions and analyzing them, and in this research, an intelligent method has been used [1–7]. Topic modeling presumes that the input text document set contains several unknown subjects that need recognition. Each subject (topic) is an unknown distribution of words, and each review (text document) is a distribution of subjects. The aim is to detect concealed knowledge in textual data related to the user’s comments. Several methods perform subject modelings, such as Latent Dirichlet Allocation (LDA) and Probabilistic Latent Semantics Analysis (PLSA). PLSA is a method that can produce the data perceived in a document-term matrix. LDA is a probabilistic method because it is exhibited in a probabilistic language, and it is a generative model because it is about ensuring that documents are produced. LDA has based on the premise that a review is a combination of subjects in which each topic is distributed over words. The linear growth of PLSA parameters indicates that the method is prone to overfitting. LDA can be easily extended to new documents. In addition, increasing the training data size does not lead to the growth of LDA-related parameters [7].

In LDA, subjects are related to documents, and words are related to subjects. To model the emotion of reviews, Joint Sentiment-Topic (JST) [8] establishes an extra layer of emotion between the layers of document and subject, where the emotion labels are related to the documents, the subjects are related to the emotion labels, and words are tagged with emotions and related topics. This study assumes that each word in a document affects its neighbors, and different words in the document may be affected simultaneously by several neighbor words. Thus, the proposed models consider the effect of words on each other. The proposed models add two parameters (weight and window) to JST. The window parameter represents the range of the effect of a word, and the weight parameter represents the strength of the effect of the word. These two parameters play an important role in better classification, as seen in the evaluation section. Using the parameters weight and window, two new methods are introduced that have revealed notable dominance over the baseline algorithms, such as JST, Topic Sentiment modeling (TS) [9], Reverse-JST (RJST) [10], and Tying-JST model (TJST) [8].

More and more improved algorithms and strategies are used to solve sentiment analysis problems. However, none of the researchers have improved the accuracy besides generating a sentiment dictionary. Different from other related studies, in this study, the proposed models improve topic-model-based sentiment classification using two parameters (weight and window). The proposed models consider the effect of words on each other. They can also generate a sentiment dictionary that includes words and scores that specify positive and negative labels and their weight. Accuracy is calculated using two formulas. Finally, by evaluating the proposed methods and the comparison with other algorithms on thirteen datasets of different sizes, the results show that the algorithms presented in this study are superior to the compared algorithms in terms of accuracy, perplexity, and topic\_coherency.

The rest of this article is arranged as follows: Section 2 shows a summarized overview of previous works in emotion analysis and the use of topic modeling in emotion analysis. The proposed models are provided in Section 3. The evaluation results are discussed in Section 4, and Section 5 concludes this article.

## 2. Related Works

The value of emotion analysis may be highlighted by analyzing customer happiness from online services like email. It is also feasible to employ emotion mining to evaluate the opinions of various people in order to make them aware of things that have favorable reviews. Major types of classification in emotion analysis are document, sentence, and aspect. An opinion is a quadrilateral  $(g, s, h, t)$ , where  $g$  is the target,  $s$  is sentiment,  $h$  is the author’s opinion, and  $t$  is the opinion expression time [11, 12, 13]. Many attempts have been made to detect emotions and explore the knowledge embedded in text data. Topic modeling obtains concealed subjects of documents. In topic modeling, the aim is to discover the best set of hidden variables that can express the observed data. LDA has been used as a topic model to effectively explore subjects in the documents [7]. LDA has motivated countless algorithms to expand to solve different problems [14–17]. In [18], the authors exhibit three topic models which make better LDA using date, helpfulness, and subtopic parameters. Articles [8, 10, 19] describe the methodology JST. This model expands LDA using a sentiment layer. This method cannot accurately identify the different emotions and is used as a baseline method in most articles. Several methods are similar to JST [8, 10, 20]). The aspect and Sentiment Unification Model (ASUM) [20] is similar to JST. JST assumes that each word represents an aspect, but ASUM assumes that each sentence represents a description of an aspect. A variation of the JST model is TJST [8]. The main difference between JST and TJST is that to sample a word in a document during the generative process of documents, JST selects a subject-document distribution for each document, whereas TJST uses one subject-document distribution for all documents. According to [10], the emotion influences the subject in JST, whereas in RJST, the subject influences the emotion. According to [9], there is only one topic-sentiment distribution for all documents in the TS, while there is a distribution for each document in RJST.

Several methods have been introduced for text emotion analysis that uses topic modeling [21–23, 78]. In [24], the authors introduce an algorithm that creates a review containing both shared subjects and subjects distributed over words as special data. Two topic models are proposed in [79]: Multilabel Supervised Topic Model (MSTM) and Sentiment Latent Topic Model (SLTM). Both methods could be used to categorize social emotions. In [25], the authors introduce a Sentiment Enriched and Latent-Dirichlet-Allocation-based review rating Prediction (SELDAP) to predict ratings using topics and sentiments of reviews. In [26], the authors introduce a method named Hierarchical Clinical Embeddings combined with Topic modeling (HCET), which can integrate five types of Electronic Health Record (EHR) data over several visits to predict depression. The authors of [80] presented the word Sense aware

LDA (SLDA) approach that uses word sense in topic formation. In [27], the authors introduce a survey of different short text topic modeling methods. They provide a detailed analysis of algorithms and discuss their performance. The authors proposed a segment-level joint topic-sentiment model (STSM) in [81], where each sentence is divided into parts by conjunctions, and the assumption that all terms in a section convey the same emotion is presented. In [28], the authors provided a thorough examination of subject modeling methods.

Deep learning provides an approach to utilizing large volumes of calculation and data using little manual engineering. Recently, deep learning approaches to analyzing emotions have reached a considerable triumph [29, 30, 47, 77]. Optimization methods have developed significantly in recent years [31–37]. Optimization methods are widely used in the feature section, notably for text. In [38], the authors proposed a multiobjective-grey wolf-optimization algorithm to categorize sentiments. In [39], the authors proposed a binary grey wolf optimizer method to classify labels in the text. In the following article [40], the authors introduced a new optimization method that mimics the model of a successful person in society. Their article used this method to categorize emotions, which achieved very good results. There are several works on using user behavior for sentiment analysis. Tag sentiment aspect (TSA) framework, a new probabilistic generative topic framework, was presented by [48] with three implementation editions. TSA is on the basis of LDA. In [41], the authors concentrate on user-based methods on social networks, where users create text data to show their views on different topics and make connections with other users to create a social network. In [42], the authors used a signed social network to detect the emotions of reviews as an unsupervised approach. Various works use other techniques for sentiment analysis problems [43–45]. In Adaptive Lexicon learning using a Genetic Algorithm (ALGA) [46], some emotion dictionaries for a dataset in the training stage are constructed using the genetic method. These sets are utilized in the testing stage. Each lexicon comprises both words and their scores. A chromosome is modeled as a vector of emotional words and scores in the genetic approach. Scores are in the range of (the lowest score of an emotional word, the highest score of an emotional word). The main goal of ALGA is to create a lexicon that minimizes the error in the training stage.

In [47], the authors proposed a deep learning-based topic-level opinion mining method. The approach is novel in that it works at the level of the sentence to explore the subject using online latent semantic indexing and then employs a subject-level attention method in an extended short-term memory network to detect emotion. In [62], the authors proposed a joint aspect-based sentiment topic model that extracts multi-grained aspects and emotions. In [49], parts-of-speech (POS) tagging is performed via a hidden Markov model, and unigrams, bigrams, and bi-tagged features are extracted. Also, the nonparametric hierarchical Dirichlet process is employed to extract the joint sentiment-topic features. In [50], the authors used an unsupervised machine learning method to extract emotion at the document and word levels. In [51], the authors proposed a new framework for joint sentiment-topic modeling based on the Restricted Boltzmann Machine (RBM), a type of

neural network. In [52], the authors proposed a probabilistic method to incorporate textual reviews and overall ratings, considering their natural connection for a joint sentiment-topic prediction. In [53], the authors proposed a hybrid topic model-based method for aspect extraction and emotion categorization of reviews. LDA is used for aspect extraction and two-layer bidirectional long short-term memory for emotion categorization. In [54], the authors proposed a joint sentiment-topic model that uses Markov Random Field Regularizer and can extract more coherent and diverse topics from short texts. In [55], the authors proposed a topic model with a new document-level latent sentiment variable for each topic, which moderates the word frequency within a topic. In [56], the authors proposed a new method for text emotion detection, aiming to improve the LSTM network by integrating emotional intelligence and attention mechanism. In [57], the authors proposed a new model for aspect-based emotion detection. The model is a novel adaptation of the LDA algorithm for product aspect extraction.

In [58], the authors introduced a new deep learning-based algorithm for emotion detection, using available ratings as weak supervision signals. In [59], the authors introduced a new deep learning-based algorithm for emotion detection, using two hidden layers. The first layer learns sentence vectors to represent the semantics of sentences, and in the second layer, the relations of sentences are encoded. In [60], the authors introduced a transformer-based model for emotion detection that encodes representation from a transformer and applies deep embedding to improve the quality of tweets. In [61], the authors introduced an attention-based deep method using two independent layers. By having to consider temporal information flow in two directions, it will retrieve both past and future contexts.

In this study, the proposed methods have tried to increase the accuracy with fewer parameters and, at the same time, simplicity compared to the existing methods. The proposed methods analyze emotions at the document-level and create an emotional dictionary. They are also the first methods that create an emotional dictionary through a topic modeling technique automatically and accurately. The proposed methods are the first methods that consider the words in the text and their effect on each other in a dynamic and weighty way.

Table 1 compares a number of articles presented in recent years in emotion analysis in terms of method, language, and dataset. In the method column, as can be seen, the combination of topic modeling and deep learning methods has recently been considered. In the language column, it is specified in which language the proposed method has been tested. The name of the dataset that has been tested can also be seen in the dataset column.

### 3. Proposed Models

This study proposes two novel topic sentiment models called Weighted Joint Sentiment-Topic (WJST) and Weighted Joint Sentiment-Topic 1 (WJST1). The proposed models improve JST using two extra parameters (weight and window).



TABLE 1: A general comparison of similar methods in recent years.

References	Method	Language	Dataset	General result
Pathak et al. [47]	Deep learning + topic modeling	English	Facebook, Ethereum, Bitcoin, SemEval-2017	Facebook-0.79, Ethereum-0.844, Bitcoin-0.817, SemEval-2017-0.889
Tang et al.[62]	Topic modeling	English	Amazon, Yelp	Amazon-0.82, Yelp-0.84
Kalarani and Selva Brunda [49]	Joint sentiment-topic features + POS tagging + SVM and ANN	English	Balanced dataset, unbalanced data	SVM-0.84, ANN-0.87
Farkhod et al. [50]	Topic modeling	English	IMDB	IMDB-F1 score-70.0
Fatemi and Safayani [51]	Topic modeling + restricted Boltzmann machine	English	20-Newsgroups (20NG), movie review (MR), multidomain sentiment (MDS)	Perplexity: MR: 406.74
Pathik and Shukla [53]	Deep learning + topic modeling	English	Yelp, Amazon, IMDB	Yelp- 0.75, Amazon-0.76, IMDB- 0.82
Sengupta et al.[54]	Topic modeling	English	Movies, Twitter	Perplexity: Movies- 3834.7, Twitter- 280.75
Huang et al.[56]	Deep learning	English	IMDB, Yelp	IMDB-0.963, Yelp-0.735
Özyurt and Akcayol [57]	Topic modeling	English + Turkish	User reviews in Turkish language about smartphones, SemEval-2016, Task-5 Turkish restaurant reviews	Precision-81.36 Recall-83.43 F-score-82.39
Zhao et al. [58]	Deep learning	English	Amazon review	CNN-87.7, LSTM-87.9
Rao et al. [59]	Deep learning	English	Yelp 2014, 2015, IMDb	Yelp2014-63.9 Yelp2015-63.8, IMDb-44.3
Naseem et al. [60]	Deep learning	English	Airline dataset	Airline dataset = 0.95
Basiri et al. [61]	Attention-based deep learning	English	Sentiment140, Airline, Kindle dataset, movie review	Kindle dataset = 0.93, Airline = 0.92, movie review = 0.90, Sentiment140 = 0.81

The proposed models are deeply described step-by-step in the next section.

According to Figure 1, the data type of the **dataset** is text. **Preprocessing** is performed by lowercasing all words, removing the stop words and words with too low and too high frequency, stemming, removing digits, and removing nonalphanumeric characters such as (#, ! . . .). Proposed models can be summarized as follows: (1) in the **Generative Model** part, the procedure of generating a word in a document under a topic model is illustrated. (2) In the **Plate Notation** part, a graphical representation of the subject model is provided (in the style of plate notation). (3) In the **Model Inference** part, Gibbs sampling will be used (to fulfill approximation inference). In the **Evaluation** phase, the model’s performance is evaluated using accuracy, perplexity, and topic\_coherency.

**3.1. Motivation.** The proposed models add two parameters to JST as latent variables in this study. From our view, it is assumed that the words in the documents affect their neighbors, and different words in the document may be affected simultaneously by several neighbor words. For example, in the sentence “My phone has a small memory, and its pictures quality is low,” the unigram *small* affects the unigram *memory*, and the bigram *small memory* affects the unigrams *phone* and *pictures*. So, unigram *small* affects unigrams *memory*, *phone*, and *picture*.

According to Figure 2, the reviews as input text data types are used for sentiment classification. The proposed models consider the effect of words on each other. They adopt Gibbs sampling to perform approximate inference of distributions. After completing the sampling in the Gibbs sampling algorithm, latent variables’ distribution can be calculated. Sentiment classification at the document-level is calculated based on the probability of a sentiment label given to a document.

Like the above example, a word can affect neighbor words in many sentences. So, in the proposed models, we consider the effect of words on each other using two parameters. The window parameter represents the range of the impact of a word, and the weight parameter represents the strength of the effect of the word. In the proposed models, each word has a weight, a sentiment label, and a topic and affects its neighbors as much as its window size, which means that each word has a window. For instance, as can be seen in Figure 3, word  $w_3$  has the window size equal to 1 and affects words  $w_2$  and  $w_4$ , and  $w_6$  has the window size equal to 2 and affects words  $w_4$ ,  $w_5$ ,  $w_7$ , and  $w_8$ . If word  $w_3$  had weight  $h$  and negative sentiment, words  $w_2$  and  $w_4$  would have weight  $h$  and negative sentiment as well. Each word is affected by its neighbors. So, different words in a document may be affected simultaneously by several neighbor words.

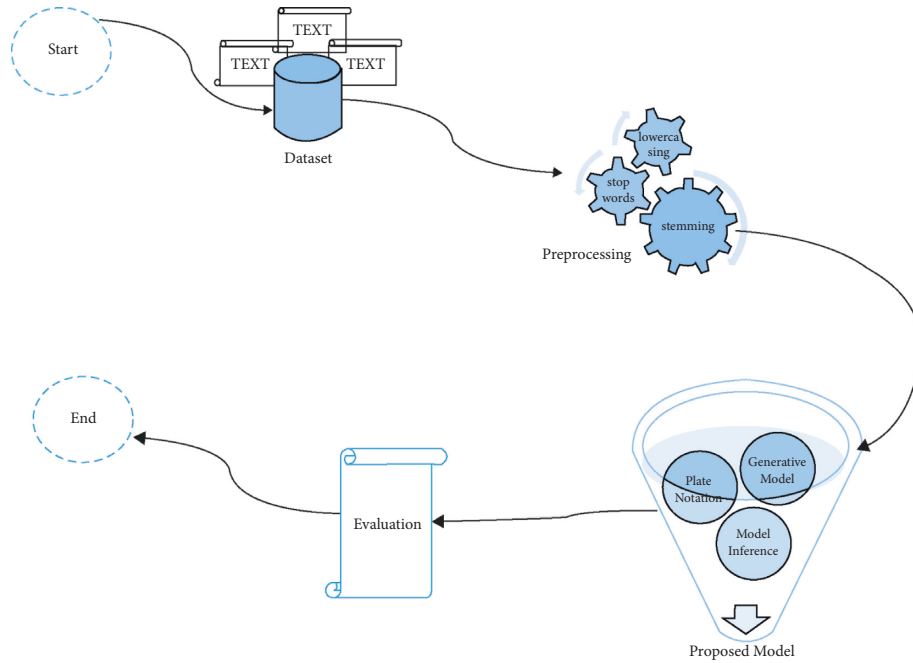


FIGURE 1: The framework chart of the proposed methods.

**3.2. The Problem Statement.** In this study, given a corpus of  $|R|$  documents,  $R = \{r_1, r_2, r_3, \dots, r_{|R|}\}$ , a document  $r$ , consists of  $\{w_1, w_2, w_3, \dots, w_{N_r}\}$  words, and each word belongs to a vocabulary set with a  $|V|$  distinct element. Furthermore,  $|Q|$  is the number of separate windows,  $|E|$  is the number of distinct weights,  $|S|$  is the number of distinct sentiment labels, and  $|Z|$  is the number of distinct topics. In the present study, five sets  $\theta, \varphi, \pi, \psi,$  and  $\xi$  require to be inferred which are latent variables. The hyperparameters  $\alpha, \beta, \gamma, \delta,$  and  $\mu$  are given based on the experience, which can be the prior observation counts before observing any actual words, where  $\alpha$  is Dirichlet prior distribution for  $\theta$ ,  $\beta$  is Dirichlet prior distribution for  $\varphi$ ,  $\gamma$  is Dirichlet prior distribution for  $\pi$ ,  $\delta$  is Dirichlet prior distribution for  $\psi$ , and  $\mu$  is Dirichlet prior distribution for  $\xi$ . The latent parameters  $\mathbf{z}, \mathbf{s}, \mathbf{q}, \mathbf{e}, \varphi, \theta, \pi, \xi,$  and  $\psi$  require to be approximated using observed variables, where  $\mathbf{z}$  is topic variable,  $\mathbf{s}$  is sentiment variable,  $\mathbf{q}$  is window variable, and  $\mathbf{e}$  is weight variable. The proposed models demonstrate the process of generating words in documents. Furthermore, they can approximate the latent variables. In the present study, the main aim of the proposed topic models is to categorize sentiments at the document-level.

**3.2.1. The Problem We Are Trying to Solve or Improve.** Analyzing user satisfaction with various services, products, or movies is the main problem in this study, mainly reflected in users' comments. A user's comment is formed by a message as text on the Internet which can be a tweet or a simple message on a website. So, for example, it is feasible to employ emotion mining to evaluate the opinions of various people in order to make them aware of things that have favorable reviews.

**3.2.2. The Solution to the Problem.** Many attempts have been made to detect emotions and explore the knowledge embedded in text data. Topic modeling as a known method can obtain concealed subjects of documents. LDA has been used as a topic model to effectively explore issues in the documents. As an LDA-based model, JST examines the impact of topics and emotions on words. The emotion parameter is insufficient, and additional parameters may play valuable roles in achieving better performance.

This study presents two new topic models that extend and improve JST through two new parameters and generate a sentiment dictionary. The proposed models consider the effect of words on each other, which, from our view, is an important factor and can increase the performance of baseline methods. Several methods have been introduced for text emotion analysis that uses topic modeling. However, none of the researchers have improved the accuracy besides generating a sentiment dictionary. Different from other related studies, in this study, the proposed models improve topic-model-based sentiment classification using two parameters (weight and window). The proposed models are deeply described step-by-step in the following sections.

**3.3. The General Structure of WJST.** This subsection introduces a new model named WJST, which improves JST using two parameters (weight and window). The primary goal of WJST is to classify sentiments at the document-level. A summary of symbols applied in the model is prepared in Table 2. The process of generating a word of a document in WJST can be outlined as follows: (1) for each document, an author first decides the distribution of sentiments. For example, sentiments are 70% positive and 30% negative, so the proposed model chooses a sentiment label from the per-

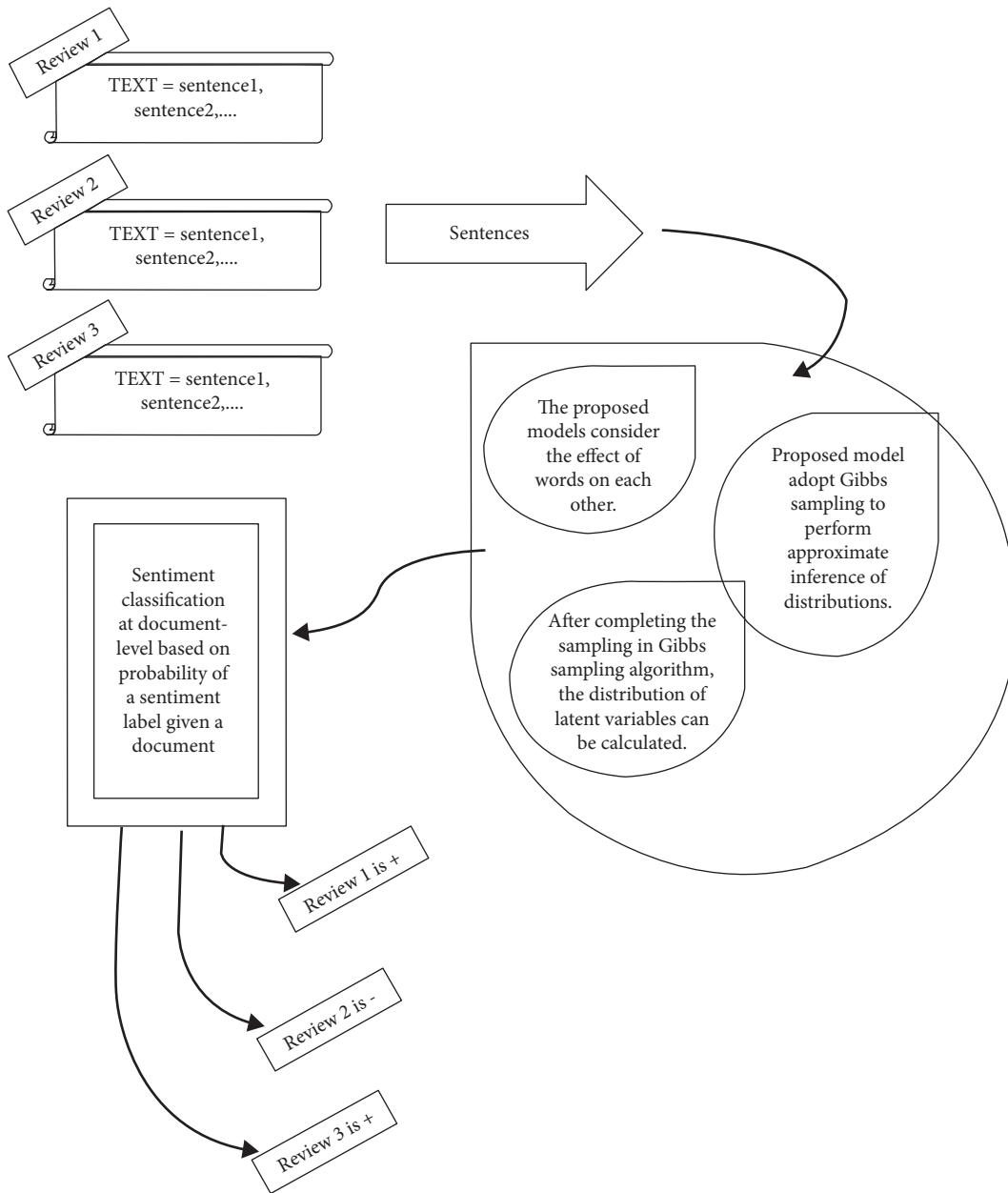


FIGURE 2: The general architecture of the proposed methods.

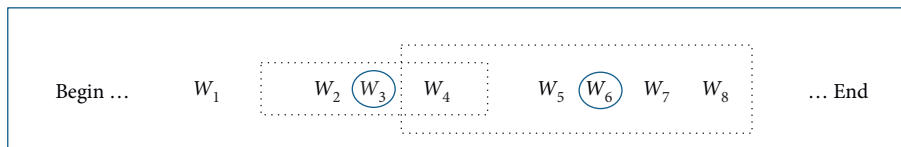


FIGURE 3: An example of a sentence with different windows.

document sentiment distribution. (2) After determining the sentiment label, the author writes a review about a product according to the distribution of topics. For example, topics are 70% about memory, 20% about speed, and 10% about battery, so WJST chooses a topic from the per-document topic distribution that depends on the sentiment label. (3) After determining the sentiment label and the topic, the

author decides the distribution of weights and the distribution of windows. WJST then chooses a weight from the per-document weight distribution that depends on the sentiment label and topic. WJST chooses a window size from the per-document window distribution that depends on the topic. (4) Finally, the author chooses some words to express an opinion under the identified topic, sentiment label,

TABLE 2: A summary of notations used in WJST.

Symbol	Description
<i>Collections</i>	
$R$	Set of all documents
$V$	Vocabulary set
$Q$	Set of all distinct windows (with different sizes)
$Z$	Set of all topics
$E$	Set of all distinct weights
$S$	Set of all sentiment labels
<i>Init parameters</i>	
$q$	Window variable
$e$	Weight variable
$r$	Document variable
$z$	Topic variable
$w$	Word variable
$s$	Sentiment variable
<i>Distributions</i>	
$\theta$	Probability of $z$ given $s$ and $r$
$\varphi$	Probability of $w$ given $z, s, q,$ and $e$
$\pi$	Probability of $s$ given $r$ ( $s_0$ = Positive label, $s_1$ = Negative label)
$\psi$	Probability of $e$ given $z, s,$ and $r$
$\xi$	Probability of $q$ given $z$ and $r$
<i>Hyper parameters</i>	
$\alpha$	Dirichlet prior distribution for $\theta$
$\beta$	Dirichlet prior distribution for $\varphi$
$\gamma$	Dirichlet prior distribution for $\pi$
$\delta$	Dirichlet prior distribution for $\psi$
$\mu$	Dirichlet prior distribution for $\xi$

weight, and window. So, WJST draws a word from the per-corpora word distribution that depends on the topic, sentiment label, weight, and window. The words with different topics may have different window sizes. For example, a word with topic *memory* has a smaller window size than a word with topic *mobile* because topic *mobile* is more general than topic *memory* which can cover topic *memory*. So, the **topic affects window size**.

The words with different topics may have different weights. For example, word *size* in topic *memory* is significant and considerable weight because all customers like memories with larger capacity. Word *size* in topic *mobile* is not important as word *size* in topic *memory*, and it has a small weight in topic *mobile* because some customers may

like mobile phones with small size (iPhone 6s), and some customers may like the mobile phones with large size (iPhone 6s+). So, the **topic affects weight**. The words with different sentiment labels may have different weights. For example, suppose that topic *memory* contains two words *size* and *cost*. If the word *size* is positive, positive size will be more important than the word *cost*, and its weight will be larger than the cost. If word *size* is negative, the positive cost will be more important than the word *size*, and its weight will be larger than the size. Positive size means using words like *large* and *big* because customers like memories with larger capacity sizes. Negative size means using words like *small* because costumers do not like memories with smaller capacity size. Positive cost means using words like *low* and *cheap* because costumers like low-priced memories. Negative cost means using words like *high* and *expensive* because costumers do not like high-priced memories. So, **sentiment label affects weight**. The proposed model is parametric in this study [63]. Furthermore, the number of topics is constant. The generative model of WJST is demonstrated in Figure 4.

The symbols of *Multi* and *Dir* demonstrate distributions of Multinomial and Dirichlet, respectively. Five sets of latent variables  $\theta, \varphi, \pi, \psi,$  and  $\xi$  require to be inferred which are latent variables. The hyperparameters  $\alpha, \beta, \gamma, \delta,$  and  $\mu$  are given based on the experience, which can be the prior observation counts before observing any actual words. The latent parameters  $z, s, q, e, \theta, \varphi, \pi, \psi,$  and  $\xi$  require to be approximated using observed variables. The plate notation of WJST is exhibited in Figure 5. The plate notation is a method for expressing variables repeating in a graphical model. Furthermore, a probabilistic model shows the conditional dependency layout among the random variables as a graph.

According to Figure 5, the joint probability distributions for the model WJST can be factored as follows:

$$P(\mathbf{w}, \mathbf{z}, \mathbf{s}, \mathbf{q}, \mathbf{e}) = P(\mathbf{w}|\mathbf{z}, \mathbf{s}, \mathbf{q}, \mathbf{e}) \times P(\mathbf{z}|\mathbf{s}, \mathbf{r}) \times P(\mathbf{s}|\mathbf{r}) \times P(\mathbf{q}|\mathbf{z}, \mathbf{r}) \times P(\mathbf{e}|\mathbf{z}, \mathbf{s}, \mathbf{r}), \quad (1)$$

where by integrating out  $\varphi$ , we achieve:

$$P(\mathbf{w}|\mathbf{z}, \mathbf{s}, \mathbf{q}, \mathbf{e}) = \left( \frac{\Gamma(|V| \times \beta)}{\Gamma(\beta)^{|V|}} \right)^{|\mathbf{Z}| \times |\mathbf{S}| \times |\mathbf{Q}| \times |\mathbf{E}|} \prod_z \prod_s \prod_q \prod_e \frac{\prod_w \Gamma(N_{w,z,s,q,e} + \beta)}{\Gamma(N_{z,s,q,e} + (|V| \times \beta))}, \quad (2)$$

where  $|V|$  is the vocabulary size,  $|S|$  is the number of sentiment labels,  $|Z|$  is the number of topics,  $|Q|$  is the number of distinct windows, and  $|E|$  is the number of weights. The symbol  $N_{w,z,s,q,e}$  is the number of times the word  $w$  has been assigned to topic  $z$ , window  $q$ , weight  $e$ , and sentiment  $s$ . The symbol  $N_{z,s,q,e}$  is the number of words with topic  $z$ , window  $q$ , weight  $e$ , and sentiment  $s$ . The symbol  $\beta$  is Dirichlet prior to  $\varphi$ . The symbol  $\Gamma$  is the gamma function. In addition, by integrating out  $\theta$ , we achieve:

$$P(\mathbf{z}|\mathbf{s}, \mathbf{r}) = \left( \frac{\Gamma(|Z| \times \alpha)}{\Gamma(\alpha)^{|Z|}} \right)^{|\mathbf{S}| \times |\mathbf{R}|} \prod_r \prod_s \frac{\prod_z \Gamma(N_{z,s,r} + \alpha)}{\Gamma(N_{s,r} + (|Z| \times \alpha))}, \quad (3)$$

where  $|R|$  is the number of documents and  $N_{z,s,r}$  is the number of words with topic  $z$  with sentiment  $s$  in document  $r$ . The symbol  $N_{s,r}$  is the number of words with sentiment  $s$  in document  $r$ . The symbol  $\alpha$  is Dirichlet before  $\theta$ . And by integrating out  $\pi$ , we achieve:

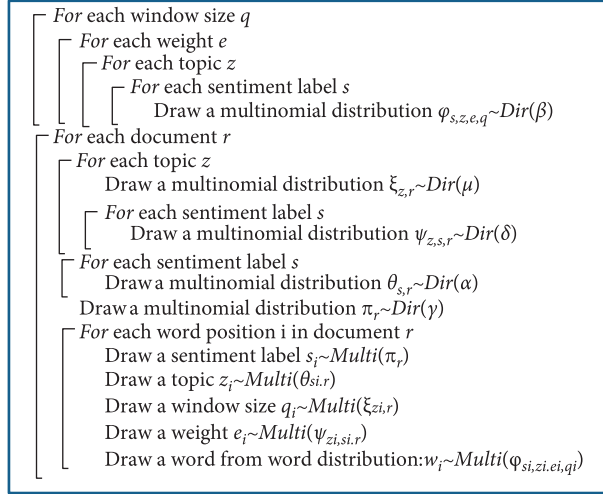


FIGURE 4: The formal definition of the process of generating words in WJST.

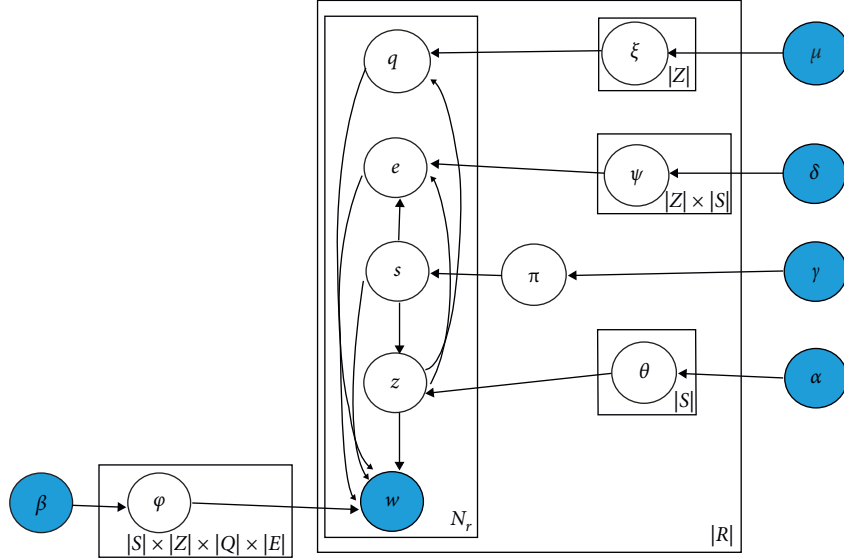


FIGURE 5: The plate notation of WJST.

$$P(\mathbf{s}|\mathbf{r}) = \left( \frac{\Gamma(|S| \times \gamma)}{\Gamma(\gamma)^{|S|}} \right)^{|\mathbf{R}|} \prod_r \frac{\prod_s \Gamma(F_{s,r} + \gamma)}{\Gamma(F_r + (|S| \times \gamma))}, \quad (4)$$

where  $F_{s,r}$  is the effect of words with sentiment  $s$  in document  $r$ , which is equal to  $\sum |e_{w,s,r} \times (1 + 2 \times q_{w,s,r})|$  where  $e_{w,s,r}$  is the weight of word  $w$  with sentiment  $s$  in document  $r$  and  $q_{w,s,r}$  is the window size of word  $w$  with sentiment  $s$  in document  $r$ . The symbol  $F_r$  is the sum of the effect of words with different sentiments (positive and negative) in document  $r$ , which is equal to  $\sum_{s \in \{\text{positive, negative}\}} F_{s,r}$ . The symbol  $\gamma$  is Dirichlet before  $\pi$ . And by integrating out  $\xi$ , we achieve:

$$P(\mathbf{q}|\mathbf{z}, \mathbf{r}) = \left( \frac{\Gamma(|Q| \times \mu)}{\Gamma(\mu)^{|Q|}} \right)^{|\mathbf{Z}| \times |\mathbf{R}|} \prod_r \prod_z \frac{\prod_q \Gamma(N_{q,z,r} + \mu)}{\Gamma(N_{z,r} + (|Q| \times \mu))}, \quad (5)$$

where  $|Q|$  is the number of distinct windows. The symbol  $N_{q,z,r}$  is the number of words with topic  $z$  and window  $q$  in document  $r$ . The symbol  $N_{z,r}$  is the number of words with topic  $z$  in document  $r$ . The symbol  $\mu$  is Dirichlet before  $\xi$ . And by integrating out  $\psi$ , we achieve:

$$P(\mathbf{e}|\mathbf{z}, \mathbf{s}, \mathbf{r}) = \left( \frac{\Gamma(|E| \times \delta)}{\Gamma(\delta)^{|E|}} \right)^{|\mathbf{Z}| \times |\mathbf{S}| \times |\mathbf{R}|} \prod_s \prod_r \prod_z \frac{\prod_e \Gamma(N_{e,z,s,r} + \delta)}{\Gamma(N_{z,s,r} + (|E| \times \delta))}, \quad (6)$$

where  $|E|$  is the number of weights,  $N_{e,z,s,r}$  is the number of words with topic  $z$ , weight  $e$ , and sentiment  $s$  in document  $r$ . The symbol  $N_{z,s,r}$  is the number of words with sentiment  $s$  and topic  $z$  in document  $r$ . The symbol  $\delta$  is Dirichlet before  $\psi$ . To estimate the parameters  $\varphi$ ,  $\theta$ ,  $\pi$ ,  $\xi$ , and  $\psi$ , we need to evaluate the above distributions. These distributions are difficult to assess directly, so we adopt Gibbs sampling to

perform approximate inference. Gibbs sampling is a widely used inference technique and is a popular approach for parameter estimation and inference in many topic models such as LDA [7]. The advantage of using the Gibbs sampling method is that it is simple and easy to implement. In this study, Gibbs sampling is used to estimate the distributions of the latent variables. The pseudocode of the Gibbs sampling

algorithm is given in Figure 6 for the proposed model, and the meanings of all variables are seen in Table 2. The algorithm will sample each variable ( $\mathbf{z}$ ,  $\mathbf{s}$ ,  $\mathbf{q}$ , and  $\mathbf{e}$ ) based on the following formula by canceling terms in equations (2)–(6) (by replacing terms in (1) with those in equations (2)–(6)):

$$P(\mathbf{z}_{r,i} = z, \mathbf{s}_{r,i} = s, \mathbf{q}_{r,i} = q, \mathbf{e}_{r,i} = e | \mathbf{z}_{r,i}, \mathbf{s}_{r,i}, \mathbf{q}_{r,i}, \mathbf{e}_{r,i}, \beta, \alpha, \gamma, \mu, \delta) \propto \frac{\{N_{w,z,s,q,e}\} + \beta}{\{N_{z,s,q,e}\} + |V| \times \beta} \times \frac{\{N_{z,s,r}\} + \alpha}{\{N_{s,r}\} + |Z| \times \alpha} \times \frac{\{F_{s,r}\} + \gamma}{\{F_r\} + |S| \times \gamma} \quad (7)$$

$$\times \frac{\{N_{q,z,r}\} + \mu}{\{N_{z,r}\} + |Q| \times \mu} \times \frac{\{N_{e,z,s,r}\} + \delta}{\{N_{z,s,r}\} + |E| \times \delta}$$

where  $\mathbf{z}_{r,i}$ ,  $\mathbf{s}_{r,i}$ ,  $\mathbf{q}_{r,i}$ , and  $\mathbf{e}_{r,i}$  are topic, sentiment, window, and weight assignments, respectively, for all the words in the collection, except for the word considered at position  $i$  in document  $r$ . Posterior inference of parameters is performed using Gibbs sampling, as demonstrated in Figure 6.

In the section of initialization, the method randomly sets the parameters. A sentiment dictionary is employed for initializing sentiment labels. The sentiment dictionary contains words and scores that specify positive and negative labels and their weight. In this study, AFINN [64] is used as a sentiment dictionary, improving the model's accuracy. At the end of the sampling algorithm, each word has a weight and a sentiment label. Therefore, a dictionary can generate sentiment scores (weights and sentiment labels) and words. The scores are extracted from a dataset based on  $P(\mathbf{w} | \mathbf{s}, \mathbf{e})$ . Each word's weight and sentiment with the most probability are selected as sentiment scores among all documents. Adaptive Lexicon learning using Genetic Algorithm (ALGA) [46] uses the genetic algorithm to generate a sentiment dictionary. However, we use topic modeling in WJST, to generate this dictionary. In WJST, the window size is different for various words. At each step of the sampling algorithm, count variables such as  $F_{s,r}$  and  $F_r$  are updated after sampling sentiment label, weight, and window size. After completing the sampling, the distribution of latent variables ( $\varphi$ ,  $\theta$ ,  $\pi$ ,  $\xi$ , and  $\psi$ ) can be calculated as follows:

$$\varphi = \frac{\{N_{w,z,s,q,e}\} + \beta}{\{N_{z,s,q,e}\} + |V| \times \beta} \quad (8)$$

$$\theta = \frac{\{N_{z,s,r}\} + \alpha}{\{N_{s,r}\} + |Z| \times \alpha} \quad (9)$$

$$\pi = \frac{\{F_{s,r}\} + \gamma}{\{F_r\} + |S| \times \gamma} \quad (10)$$

$$\xi = \frac{\{N_{q,z,r}\} + \mu}{\{N_{z,r}\} + |Q| \times \mu} \quad (11)$$

$$\psi = \frac{\{N_{e,z,s,r}\} + \delta}{\{N_{z,s,r}\} + |E| \times \delta}$$

The probability of a word given a topic would be equal to  $\sum P(\mathbf{w} | \mathbf{z}, \mathbf{s}, \mathbf{q}, \mathbf{e})$ , and the probability of a sentiment label given a document for sentiment classification at the document-level is calculated using  $\pi$ .

The time complexity of the proposed method quantifies the amount of time taken by the Gibbs sampling algorithm to run as the main function. Given the number of words in all documents  $w_{\text{ALL}}$  ( $w_{\text{ALL}} = \sum_{r \in R} N_r$ , where  $N_r$  is the number of words in document  $r$ ), the number of topics  $|Z|$ , the number of distinct windows  $|Q|$ , the number of weights  $|E|$ , and the total number of sentiment labels  $|S|$ , the time complexity of each Gibbs sampling iteration would be  $O(w_{\text{ALL}} \cdot |S| \cdot |Z| \cdot |Q| \cdot |E|)$ . Furthermore, given the number of iterations  $G$ , the total time complexity of WJST would be  $O(G \cdot w_{\text{ALL}} \cdot |S| \cdot |Z| \cdot |Q| \cdot |E|)$ . Table 3 compares different methods in terms of time complexity.

**3.4. The General Structure of WJST1.** A version of WJST called WJST1 is presented in Figure 7. The distributions  $\theta$ ,  $\xi$ , and  $\psi$  in WJST depend on the document, but in WJST1, the distributions  $\theta$ ,  $\xi$ , and  $\psi$  do not rely on the document. Dependency between documents of a domain is more than documents in different domains. A pattern in documents of a domain may not exist in documents of other domains. So, calculations on multidomain datasets should be local and not cover all domains. For example, considering the distributions  $P(\mathbf{z} | \mathbf{s})$  and  $P(\mathbf{z} | \mathbf{s}, \mathbf{r})$ , where  $\mathbf{z}$  is topic,  $\mathbf{s}$  is sentiment, and  $\mathbf{r}$  documents, in the first state  $P(\mathbf{z} | \mathbf{s})$ , topic depends on sentiment. The distribution covers all documents in different domains. Perhaps a topic is positive in one domain and negative in another domain. So, it is better to depend the topic on the documents of a domain, not all domains. Thus, the topic is limited to the document (and

```

-Input: Preprocessed text data (set of all documents); Number of topics; Dirichlet priors for distributions.
-Output:  $\varphi$ ,  $\theta$ ,  $\pi$ ,  $\xi$ , and  $\psi$ .
-Initialize init-parameters randomly for all words in a set of documents.
  For iteration=1 to max-iteration
    For each document  $r$  in the corpus  $R$ 
      For each word  $w$  in the document  $r$ 
        Exclude the word  $w$  associated with  $\mathbf{q}$ ,  $\mathbf{z}$ ,  $\mathbf{s}$ ,  $\mathbf{e}$  from count variables
        Sample parameters (weight, window size, sentiment label, and topic) using Eq. 7
        Update count variables
      End for
    End for
    if (iteration mod 100 == 0)
      Update  $\varphi$ ,  $\theta$ ,  $\pi$ ,  $\xi$ , and  $\psi$  with Eqs. 8, 9, 10, 11, and 12, respectively
    End for
  End for

```

FIGURE 6: Adopted Gibbs sampling for WJST1.

TABLE 3: The time complexity of different models.

Model	Time complexity
JST, RJST, TJST, and TS	$O(G \cdot w_{\text{ALL}} \cdot  S  \cdot  Z )$
WJST	$O(G \cdot w_{\text{ALL}} \cdot  S  \cdot  Z  \cdot  Q  \cdot  E )$

domain), and contradiction between different domains is eliminated. So, WJST is suitable for multidomain datasets, and WJST1 is a version of WJST suitable for single-domain datasets. According to Figure 7,  $\xi$  is the probability of  $\mathbf{q}$  given  $\mathbf{z}$ ,  $\theta$  is the probability of  $\mathbf{z}$  given  $\mathbf{s}$ , and  $\psi$  is the probability of  $\mathbf{e}$  given  $\mathbf{z}$  and  $\mathbf{s}$ , and the joint probability distribution for WJST1 can be factored as follows:

$$P(\mathbf{w}, \mathbf{z}, \mathbf{s}, \mathbf{q}, \mathbf{e}) = P(\mathbf{w}|\mathbf{z}, \mathbf{s}, \mathbf{q}, \mathbf{e}) \times P(\mathbf{z}|\mathbf{s}) \times P(\mathbf{s}|\mathbf{r}) \times P(\mathbf{q}|\mathbf{z}) \times P(\mathbf{e}|\mathbf{z}, \mathbf{s}), \quad (12)$$

where by integrating out  $\theta$ , we achieve:

$$P(\mathbf{z}|\mathbf{s}) = \left( \frac{\Gamma(|Z| \times \alpha)}{\Gamma(\alpha)^{|Z|}} \right)^{|S|} \prod_s \frac{\prod_z \Gamma(N_{z,s} + \alpha)}{\Gamma(N_s + (|Z| \times \alpha))}, \quad (13)$$

where  $N_{z,s}$  is the number of words with topic  $z$  and sentiment  $s$ . The symbol  $N_s$  is the number of words with sentiment  $s$ . The symbol  $\alpha$  is Dirichlet before  $\theta$ . And by integrating out  $\xi$ , we achieve:

$$P(\mathbf{q}|\mathbf{z}) = \left( \frac{\Gamma(|Q| \times \mu)}{\Gamma(\mu)^{|Q|}} \right)^{|Z|} \prod_z \frac{\prod_q \Gamma(N_{q,z} + \mu)}{\Gamma(N_z + (|Q| \times \mu))}, \quad (14)$$

where  $N_{q,z}$  is the number of words with topic  $z$  and window  $q$ . The symbol  $N_z$  is the number of words with topic  $z$ . The symbol  $\mu$  is Dirichlet before  $\xi$ . And by integrating out  $\psi$ , we achieve:

$$P(\mathbf{e}|\mathbf{z}, \mathbf{s}) = \left( \frac{\Gamma(|E| \times \delta)}{\Gamma(\delta)^{|E|}} \right)^{|Z| \times |S|} \prod_s \prod_z \frac{\prod_e \Gamma(N_{e,z,s} + \delta)}{\Gamma(N_{z,s} + (|E| \times \delta))}, \quad (15)$$

where  $N_{e,z,s}$  is the number of words with topic  $z$ , weight  $e$ , and sentiment  $s$ . The symbol  $N_{z,s}$  is the number of words with sentiment  $s$  and topic  $z$ . The symbol  $\delta$  is Dirichlet before  $\psi$ . The symbols  $P(\mathbf{w}|\mathbf{z}, \mathbf{s}, \mathbf{q}, \mathbf{e})$  and  $P(\mathbf{s}|\mathbf{r})$  are

calculated using equations (2) and (4), respectively. After completing the sampling, the distribution of latent variables ( $\theta$ ,  $\xi$ , and  $\psi$ ) is calculated as follows:

$$\begin{aligned} \theta &= \frac{\{N_{z,s}\} + \alpha}{\{N_s\} + |Z| \times \alpha}, \\ \xi &= \frac{\{N_{q,z}\} + \mu}{\{N_z\} + |Q| \times \mu}, \\ \psi &= \frac{\{N_{e,z,s}\} + \delta}{\{N_{z,s}\} + |E| \times \delta}. \end{aligned} \quad (16)$$

And  $\varphi$  and  $\pi$  are computed through equations (8) and (10), respectively. Experimental results are demonstrated in the next section.

#### 4. Experimental Results

The present study executes the methods on a computer with an Intel Core i7 CPU and 8 GB RAM. Proposed models are compared on 13 datasets. 4 datasets crawled from Amazon (<https://www.amazon.com>) opinions include Electronic, Movie, Android, and Automotive. 2 MDS datasets [65] contain Magazines and Sports. A dataset crawled from the IMDB movie archive [3] is MR. 3 UCI datasets [66] include Amazon, Yelp, and IMDB. 3 Twitter datasets [46] include STS-Test, SOMD, and Sanders. Data preprocessing contains (1) lowercasing all words, (2) removing digits, nonalphanumeric characters, stop words, and words with too low and too high frequency, and (3) stemming. The details of the datasets are provided in Table 4.

The number of topics is unknown, provided as a constant amount at the beginning of the Gibbs sampling algorithm. In this study,  $\alpha$ ,  $\gamma$ ,  $\beta$ , and  $\delta$  specific distributions are symmetric, and we empirically set the value of parameters, and this setting demonstrates fairly good performance in our experiments. Table 5 exhibits the initialization of parameters used in different algorithms.

A sentiment dictionary is employed for initializing sentiment labels. Sentiment dictionaries such as AFINN [64], IMDB [67], 8-K [67], and Bing Liu [68, 69] contain words and scores that specify positive and negative labels as

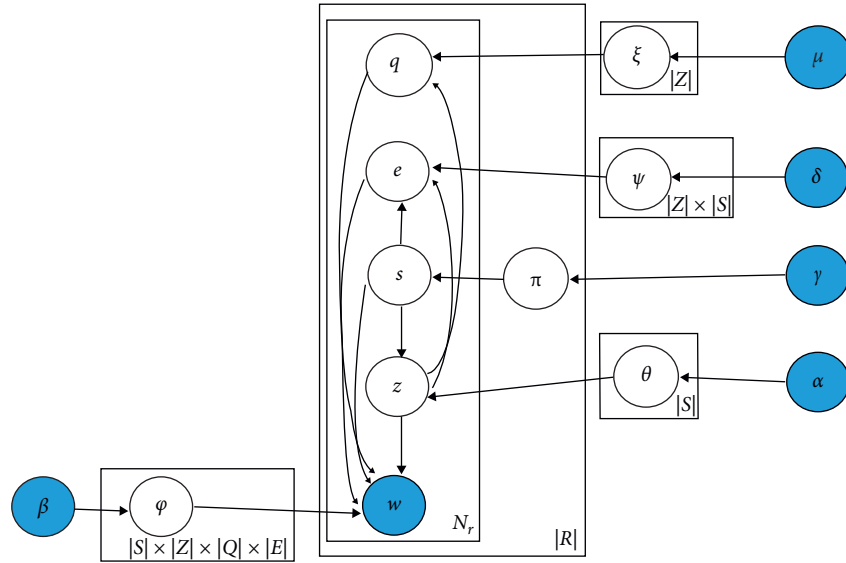


FIGURE 7: The graphical model of WJST1.

TABLE 4: Description of datasets.

#	Dataset	Number of reviews	Vocabulary size	Number of words
1	Movie	400	6592	41540
2	Electronic	400	4501	29117
3	Automotive	400	3590	19733
4	Android	400	2173	9723
5	STS	359	1489	3784
6	SOMD	916	2013	7772
7	Sanders	1224	3221	14100
8	Magazines	1800	8040	125387
9	Sports	2000	8582	113921
10	MR	2000	33054	733022
11	Amazon	1000	1521	7296
12	IMDB	1000	2556	9706
13	Yelp	1000	1679	7726

TABLE 5: Initial values of parameters.

Model	Parameters
JST	Max_iteration:5000; $ Z  = 5, 10, 15, 20$ ; $\alpha = 0.1$ ; $\gamma = 0.016 \times (\text{average document length})$ ; $\beta = 0.01$ ;
RJST	Max_iteration:5000; $ Z  = 5, 10, 15, 20$ ; $\alpha = 0.1$ ; $\gamma = 0.016 \times (\text{average document length})$ ; $\beta = 0.01$ ;
TJST	Max_iteration:5000; $ Z  = 5, 10, 15, 20$ ; $\alpha = 0.1$ ; $\gamma = 0.016 \times (\text{average document length})$ ; $\beta = 0.01$ ;
TS	Max_iteration:5000; $ Z  = 5, 10, 15, 20$ ; $\alpha = 0.1$ ; $\gamma = 0.016 \times (\text{average document length})$ ; $\beta = 0.01$ ;
WJST	Max_iteration:5000; $ Z  = 5, 10, 15, 20$ ; $\alpha = 0.3$ ; $\gamma = 0.016 \times (\text{average document length})$ ; $\beta = 0.01$ ; $\mu = 3$ ; $\delta = 9$ ; $E = [-5, +5]$ ; $Q = \{1, 2, 3, 4, 5, 6\}$ ;
WJST1	Max_iteration:5000; $ Z  = 5, 10, 15, 20$ ; $\alpha = 0.3$ ; $\gamma = 0.016 \times (\text{average document length})$ ; $\beta = 0.01$ ; $\mu = 3$ ; $\delta = 9$ ; $E = [-5, +5]$ ; $Q = \{1, 2, 3, 4, 5, 6\}$ ;

well as their weight. In the present study, AFINN is used as a sentiment dictionary which improves the model's accuracy. Sentiment detection at the document-level, perplexity, and topic\_coherency are used to compare the efficacy of proposed models as three standard parameters which are used in different papers [7, 70, 71–73].

In the present study, the Accuracy parameter uses the formula of  $((TP + TN)) / ((TP + FP + TN + FN))$ , where TP is

the number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives.

$\pi$  distribution equation (10) determines how likely each comment is positive or negative. For example, if the value of  $P(+)$  is more significant than the value of  $P(-)$  (for a comment), the comment will be positive. The Accuracy's formula uses  $\pi$  distribution (equation (10)) to calculate TP,



TABLE 6: Sentiment classification on Android, Automotive, Electronic, and Movie datasets.

Metric\ model	Android								
	RND	AFINN	RND + AFINN	JST	TJST	RJST	TS	WJST	WJST1
Accuracy1	0.48	0.6975	0.58	0.625	0.765	0.5825	0.5425	0.795	0.865
Accuracy2	—	—	—	—	—	—	—	0.7825	0.8525
Perplexity	—	—	—	17.4581	19.7185	17.4426	17.8706	14.396	14.7631
Topic_coherency	—	—	—	-2.0645	-0.8536	-1.9914	-2.373	-0.5547	-0.187
<i>Automotive</i>									
Accuracy1	0.4925	0.625	0.535	0.6575	0.7675	0.615	0.5525	0.755	0.8
Accuracy2	—	—	—	—	—	—	—	0.7475	0.795
Perplexity	—	—	—	22.6838	24.0385	21.8044	22.4878	18.4612	19.0627
Topic_coherency	—	—	—	-1.0158	-0.4712	-1.4986	-0.9008	-0.9311	-0.326
<i>Electronic</i>									
Accuracy1	0.465	0.675	0.52	0.7025	0.76	0.5525	0.5475	0.8625	0.8475
Accuracy2	—	—	—	—	—	—	—	0.875	0.855
Perplexity	—	—	—	23.3586	24.3024	23.471	24.0239	19.2999	20.2452
Topic_coherency	—	—	—	-1.5892	-1.0482	-1.2996	-1.2719	-0.5322	-1.1683
<i>Movie</i>									
Accuracy1	0.525	0.595	0.555	0.7575	0.9475	0.62	0.5425	0.8475	0.97
Accuracy2	—	—	—	—	—	—	—	0.8325	0.9675
Perplexity	—	—	—	25.2787	26.5813	25.1684	25.4488	21.0494	22.1082
Topic_coherency	—	—	—	-0.4089	-0.111	-1.0947	-1.0214	-0.0602	-0.1329

TN, FP, and FN values. For example, if a comment is positive and detected as positive (by the proposed methods), a unit is added to TP.

So, sentiment analysis (sentiment detection) at the document-level is realized using  $\pi$  distribution (equation (10)), and the formula of  $((TP + TN)/((TP + FP + TN + FN)))$  is used to compute the Accuracy.

The error formula can be calculated using the formula of  $(1 - \text{Accuracy})$ . Accuracy, perplexity, and topic\_coherency are used for evaluations in the present study. Further study can investigate more parameters such as MSE, MAE, and RMSE for future research.

Furthermore, Better methods have lower perplexity and also higher topic\_coherency. Given a test dataset  $D_{\text{Test}}$ , the perplexity is computed through

$$\text{Perplexity}(D_{\text{Test}}) = \exp\left(\frac{-\sum_{r=1}^{|\mathbf{r}|} \log P(w_r)}{\sum_{r=1}^{|\mathbf{r}|} N_r}\right), \quad (17)$$

where  $w_r$  are the words in document  $\mathbf{r}$ ,  $N_r$  is the length of document  $\mathbf{r}$ , and  $P(w_r)$  is the probability of words in document  $\mathbf{r}$ . The lower value of the formula over a held-out document demonstrates Better generalization efficacy. The evaluation results are shown in Tables 6–8, 9–14, and the proposed models demonstrate better results. In the report of Tables 6–8, 9–14, the perplexity of proposed methods is lower than that of baseline models. In the report of Tables 9–12, the perplexity is reduced with an increase in topics. Topic\_coherency is also calculated using

$$\begin{aligned} \text{Average Topic - Coherency}(Z) &= \frac{\left(\sum_{i=1}^{|Z|} C(V^{(z_i)})\right)}{|Z|} \\ &= \frac{\left(\sum_{i=1}^{|Z|} \sum_{m=2}^M \sum_{n=1}^{m-1} \log\left(\text{CODF}\left(v_m^{(z_i)}, v_n^{(z_i)}\right) + 1/\text{DF}\left(v_n^{(z_i)}\right)\right)\right)}{|Z|}. \end{aligned} \quad (18)$$

where  $V^{(z_i)} = (v_1^{(z_i)}, \dots, v_M^{(z_i)})$  is the list of  $M$  words that have a high probability in the topic  $z_i$ ,  $C(V^{(z_i)})$  is topic\_coherency for the topic  $z_i$ ,  $Z$  is the set of all topics,  $|Z|$  is the number of distinct topics,  $DF$  is the document frequency, and  $CODF$  is the co-occurrence of two words in different documents. A smoothing count of 1 is included to avoid taking the logarithm of zero. In the present study, topic\_coherency is computed through (18), equal to the average of topic\_coherency values in  $Z$ . Furthermore, a higher value of topic\_coherency reflects the better quality of the detected

topics.  $M$  is equal to 10, and results are demonstrated in Tables 6–8, 9–14. A different number of topics (5, 10, 15, and 20) and different distinct windows (1, 2, 3, 4, 5, and 6) are applied for evaluating models. In this part, baseline methods include JST [8], RJST [10], TJST [8], and TS [9]. In the present section, the Friedman test [74, 75] is used to examine the achievements of the comparison methods. The Friedman test is a nonparametric multiple comparison test utilized to examine the differences between algorithms by assigning the lowest rank to the best approach in minimization problems

TABLE 7: Sentiment classification on Magazine, Sport, MR, Amazon, IMDB, and Yelp datasets.

Metric\ model	Magazine								
	RND	AFINN	RND + AFINN	JST	TJST	RJST	TS	WJST	WJST1
Accuracy1	0.515	0.6522	0.5822	0.6705	0.705	0.5411	0.5022	0.8355	0.81
Accuracy2	—	—	—	—	—	—	—	0.8372	0.8083
Perplexity	—	—	—	21.8506	23.0349	21.4593	21.3828	19.9914	21.2095
Topic_coherency	—	—	—	-0.0548	-0.0348	-0.0946	-0.0561	-0.132	-0.0077
<i>Sport</i>									
Accuracy1	0.5285	0.686	0.5725	0.653	0.709	0.5565	0.5155	0.798	0.802
Accuracy2	—	—	—	—	—	—	—	0.782	0.795
Perplexity	—	—	—	22.874	23.1356	22.0264	22.3361	21.968	21.4821
Topic_coherency	—	—	—	-0.2234	-0.0876	-0.1369	-0.0544	-0.1406	-0.2242
<i>MR</i>									
Accuracy1	0.4895	0.601	0.5455	0.613	0.62	0.51	0.5	0.821	0.8445
Accuracy2	—	—	—	—	—	—	—	0.818	0.843
Perplexity	—	—	—	33.8663	35.0695	35.2359	34.6704	33.222	33.7698
Topic_coherency	—	—	—	-0.021	-0.0139	-0.0012	-0.0409	-0.001	-0.0106
<i>Amazon</i>									
Accuracy1	0.491	0.731	0.574	0.611	0.645	0.609	0.54	0.779	0.796
Accuracy2	—	—	—	—	—	—	—	0.829	0.798
Perplexity	—	—	—	12.6442	13.7316	13.349	14.2397	10.9211	12.9946
Topic_coherency	—	—	—	-0.8318	-4.3775	-0.4224	-0.5411	-0.5874	-0.2696
<i>IMDB</i>									
Accuracy1	0.498	0.698	0.575	0.605	0.616	0.546	0.545	0.76	0.77
Accuracy2	—	—	—	—	—	—	—	0.761	0.774
Perplexity	—	—	—	21.1053	20.4419	20.8543	19.7124	14.6719	18.9035
Topic_coherency	—	—	—	-1.3334	-1.4868	-0.8853	-1.1246	-0.9666	-0.9438
<i>Yelp</i>									
Accuracy1	0.506	0.689	0.559	0.579	0.614	0.561	0.547	0.737	0.773
Accuracy2	—	—	—	—	—	—	—	0.726	0.769
Perplexity	—	—	—	15.4565	16.7169	15.5965	15.3614	12.2145	13.3453
Topic_coherency	—	—	—	-2.1865	-2.5609	-1.9632	-1.1714	-2.0815	-2.3715

and the highest rank to the best approach in maximization problems.

There are several methods for the validation of classification and topic modeling-based problems. Still, the methods used in this study are the most common and are used in most articles related to our article for evaluation. Also, there are various methods for validation that we will try to use in a future study to evaluate the proposed methods. The following is the reason for choosing the validation methods used in this study:

We chose accuracy, perplexity, and coherence score as evaluation metrics because of their popularity in classification and topic modeling problems. Perplexity is an essential metric that, in theory, represents how well a model behaved on unseen data and is provided using the normalized log-likelihood technique. Meanwhile, the coherence score measures the degree of semantic similarity between high-scoring words and helps distinguish the semantical interpretation of topics based on statistical inference.

The main question we want to answer is whether the proposed methods can improve the performance of text sentiment classification. This study compares proposed methods with different baselines, including JST and recently representative approaches. Consider a Confusion Matrix for a classification problem that predicts whether a comment has positive sentiment or not. The total number of correctly

detected cases is one of the more obvious measures. When all of the classes are equally important, it is typically utilized. When True positives and True negatives are more significant, accuracy is employed. According to the accuracy criterion, one can immediately know whether the model is adequately trained or not and how it works in general. The most popular measurement for classification issues is accuracy, which is the proportion of correctly predicted cases to all cases. This metric's opposite, or error, can be calculated as 1-accuracy. In machine learning, an accuracy parameter is an excellent option for sentiment classification when the classes in the dataset are almost evenly distributed. Also, we will try to use various metrics such as recall and precision in future studies to evaluate the proposed methods.

We use the Friedman test to compare the results produced by the proposed methods and the competitors to verify the classification performance. Friedman's test is used to examine the achievements of the comparison methods. The Friedman test is a nonparametric multiple comparison test that is utilized to explore the differences between algorithms by assigning the lowest rank to the best approach in minimization problems and the highest rank to the best approach in maximization problems.

Topic modeling is one of the most important NLP fields. It aims to explain a textual dataset by decomposing it into two distributions: topics and words. A topic modeling

TABLE 8: Sentiment classification on different datasets based on different situations (AFINN and NO\_AFINN).

Metric	Metric\Dic	Android	
		AFINN	NO_AFINN
WJST	Accuracy1	0.7425	0.5725
	Accuracy2	0.7375	0.58
	Perplexity	15.53	16.1551
	Topic_Coh	-2.2654	-1.7346
WJST1	Accuracy1	0.855	0.81
	Accuracy2	0.8475	0.8075
	Perplexity	16.3399	16.0482
	Topic_Coh	-2.2228	-0.1295
<i>Automotive</i>			
WJST	Accuracy1	0.7125	0.6025
	Accuracy2	0.7025	0.6075
	Perplexity	20.4488	20.4065
	Topic_Coh	-3.2282	-1.6628
WJST1	Accuracy1	0.7925	0.7025
	Accuracy2	0.79	0.7125
	Perplexity	20.5213	21.1296
	Topic_Coh	-0.326	-1.1809
<i>Electronic</i>			
WJST	Accuracy1	0.8525	0.705
	Accuracy2	0.8425	0.6825
	Perplexity	20.0579	20.2615
	Topic_Coh	-0.5322	-0.5926
WJST1	Accuracy1	0.8475	0.76
	Accuracy2	0.855	0.765
	Perplexity	21.8195	21.5739
	Topic_Coh	-1.5586	-1.6968
<i>Movie</i>			
WJST	Accuracy1	0.8475	0.71
	Accuracy2	0.8325	0.715
	Perplexity	22.4588	22.6124
	Topic_Coh	-0.9342	-0.4637
WJST1	Accuracy1	0.9575	0.485
	Accuracy2	0.945	0.4875
	Perplexity	23.5662	22.8134
	Topic_Coh	-1.2359	-1.3717

algorithm is a mathematical or statistical model used to infer what the issues that better represent the data are. Human judgment-based review techniques can yield good results but are expensive and time-consuming. Human judgment is also not well defined.

In contrast, the appeal of quantitative metrics such as perplexity is the ability to standardize, automate, and scale the evaluation of topic models. In natural language processing, perplexity is a traditional metric for evaluating topic models. The lower value of the formula over a held-out document demonstrates better generalization efficacy.

Perplexity's inability to capture context and the relationships between words within a topic or across topics within a document is one of its drawbacks. For human understanding, semantic context is important. Approaches like topic coherency have been designed to tackle this problem by capturing the context between words in a subject. Extracting topic words is one of the main tasks in topic modeling. In most articles about topic modeling, topic\_coherency is shown as a number that represents the overall topics' interpretability and is used to assess the topics'

quality. The higher the topic\_coherency value, the better the quality of the subjects extracted.

*4.1. Sentiment Scores for the Words in a Dataset.* In this section, a dictionary is generated, including sentiment scores (weights and sentiment labels) and words. The scores are extracted from datasets based on  $P(\mathbf{w} | \mathbf{s}, \mathbf{e})$ . The weight and sentiment with the most probability are selected for each word as a sentiment score. The extracted scores for some phrases in the form of unigram can be seen in Tables 15 and 16. ALGA [46] uses the genetic algorithm to generate a sentiment dictionary; however, we use topic modeling in the proposed models to create this dictionary. According to Tables 15 and 16, ten words from each dataset are selected and scored by the proposed models. For example, the word *nice* obtains a score of 4 in WJST and obtains a score of 5 in WJST1. The scores are different in the proposed methods; for example, the word *serious* achieves a score of 1 in WJST and a score of -2 in WJST1. Table 15 is related to Android, Automotive, Electronic, and Movie datasets. Table 16 is associated with STS, Sanders, and SOMD datasets.

TABLE 9: Sentiment classification on the Android dataset according to the different number of topics.

Model	Metric\topic	5	10	15	20
RND	Accuracy	0.48	0.48	0.48	0.48
AFINN	Accuracy	0.6975	0.6975	0.6975	0.6975
AFINN + RND	Accuracy	0.58	0.58	0.58	0.58
Bing_Liu	Accuracy	0.6975	0.6975	0.6975	0.6975
Bing_Liu + RND	Accuracy	0.5775	0.5775	0.5775	0.5775
IMDB	Accuracy	0.7025	0.7025	0.7025	0.7025
IMDB + RND	Accuracy	0.6125	0.6125	0.6125	0.6125
8K	Accuracy	0.5425	0.5425	0.5425	0.5425
8K + RND	Accuracy	0.515	0.515	0.515	0.515
JST	Accuracy	0.625	0.6175	0.6225	0.6125
	Perplexity	19.6726	19.5187	19.182	17.4581
	Topic_Coh	-4.6026	-2.5848	-2.2753	-2.0645
TJST	Accuracy	0.7575	0.7175	0.765	0.7475
	Perplexity	21.1487	20.3726	20.1516	19.7185
	Topic_Coh	-0.8536	-1.568	-3.4285	-2.8739
RJST	Accuracy	0.5825	0.54	0.555	0.5325
	Perplexity	19.9429	19.1915	18.137	17.4426
	Topic_Coh	-3.3792	-1.9914	-3.403	-3.4386
TS	Accuracy	0.5425	0.5275	0.53	0.5175
	Perplexity	20.4934	19.1762	18.3833	17.8706
	Topic_Coh	-3.9618	-3.2137	-2.373	-2.569
WJST	Accuracy1	0.7925	0.7425	0.795	0.79
	Accuracy2	0.7775	0.7375	0.775	0.7825
	Perplexity	16.7303	15.53	14.6351	14.396
WJST1	Topic_Coh	-0.5547	-2.2654	-2.9122	-2.6465
	Accuracy1	0.81	0.855	0.865	0.85
	Accuracy2	0.7925	0.8475	0.8525	0.8375
WJST1	Perplexity	16.6787	16.3399	15.6662	14.7631
	Topic_Coh	-0.187	-2.2228	-1.5703	-2.0204

TABLE 10: Sentiment classification on the Automotive dataset according to the different number of topics.

Model	Metric\topic	5	10	15	20
RND	Accuracy	0.4925	0.4925	0.4925	0.4925
AFINN	Accuracy	0.625	0.625	0.625	0.625
AFINN + RND	Accuracy	0.535	0.535	0.535	0.535
Bing_Liu	Accuracy	0.64	0.64	0.64	0.64
Bing_Liu + RND	Accuracy	0.535	0.535	0.535	0.535
IMDB	Accuracy	0.59	0.59	0.59	0.59
IMDB + RND	Accuracy	0.4825	0.4825	0.4825	0.4825
8K	Accuracy	0.5025	0.5025	0.5025	0.5025
8K + RND	Accuracy	0.48	0.48	0.48	0.48
JST	Accuracy	0.6275	0.6575	0.6325	0.5975
	Perplexity	24.7154	23.6961	23.27	22.6838
	Topic_Coh	-1.486	-2.3443	-1.1354	-1.0158
TJST	Accuracy	0.76	0.7375	0.7675	0.7275
	Perplexity	25.5637	25.0633	24.5749	24.0385
	Topic_Coh	-0.59	-0.4712	-1.6252	-1.9377
RJST	Accuracy	0.615	0.55	0.54	0.535
	Perplexity	25.2654	23.8316	22.3905	21.8044
	Topic_Coh	-1.5779	-1.4986	-2.0833	-2.9936
TS	Accuracy	0.5425	0.5375	0.5525	0.55
	Perplexity	25.0705	23.8504	22.9364	22.4878
	Topic_Coh	-2.5941	-0.9008	-6.2652	-4.6902
WJST	Accuracy1	0.755	0.7125	0.74	0.745
	Accuracy2	0.7475	0.7025	0.745	0.735
	Perplexity	21.2008	20.4488	18.7049	18.4612
WJST1	Topic_Coh	-1.6542	-3.2282	-1.6783	-0.9311
	Accuracy1	0.80	0.7925	0.7925	0.7925
	Accuracy2	0.79	0.79	0.79	0.795
WJST1	Perplexity	21.4357	20.5213	20.0481	19.0627
	Topic_Coh	-1.1883	-0.326	-1.084	-0.8349

TABLE 11: Sentiment classification on the Electronic dataset according to the different number of topics.

Model	Metric\topic	5	10	15	20
RND	Accuracy	0.465	0.465	0.465	0.465
AFINN	Accuracy	0.675	0.675	0.675	0.675
AFINN + RND	Accuracy	0.52	0.52	0.52	0.52
Bing_Liu	Accuracy	0.695	0.695	0.695	0.695
Bing_Liu + RND	Accuracy	0.535	0.535	0.535	0.535
IMDB	Accuracy	0.6375	0.6375	0.6375	0.6375
IMDB + RND	Accuracy	0.5475	0.5475	0.5475	0.5475
8K	Accuracy	0.5075	0.5075	0.5075	0.5075
8K + RND	Accuracy	0.5075	0.5075	0.5075	0.5075
JST	Accuracy	0.675	0.7025	0.6475	0.6275
	Perplexity	25.227	24.6719	24.6115	23.3586
	Topic_Coh	-1.5892	-4.1751	-2.6007	-2.1239
TJST	Accuracy	0.75	0.74	0.73	0.76
	Perplexity	25.2091	24.8028	24.4092	24.3024
	Topic_Coh	-1.0482	-1.8084	-1.6694	-1.8297
RJST	Accuracy	0.5525	0.55	0.5375	0.53
	Perplexity	25.3113	24.1779	23.8378	23.471
	Topic_Coh	-1.2996	-1.3991	-4.472	-5.4363
TS	Accuracy	0.54	0.5375	0.5475	0.515
	Perplexity	26.2295	24.885	24.5048	24.0239
	Topic_Coh	-1.2719	-1.6586	-2.5403	-3.2174
WJST	Accuracy1	0.8625	0.8525	0.7675	0.675
	Accuracy2	0.875	0.8425	0.755	0.665
	Perplexity	20.5489	20.0579	19.9009	19.2999
WJST1	Topic_Coh	-2.0442	-0.5322	-1.3702	-1.0887
	Accuracy1	0.79	0.8475	0.8075	0.8275
	Accuracy2	0.80	0.855	0.8125	0.8225
WJST1	Perplexity	22.6012	21.8195	20.8551	20.2452
	Topic_Coh	-1.1683	-1.5586	-1.6646	-1.4681

TABLE 12: Sentiment classification on the Movie dataset according to the different number of topics.

Model	Metric\topic	5	10	15	20
RND	Accuracy	0.525	0.525	0.525	0.525
AFINN	Accuracy	0.595	0.595	0.595	0.595
AFINN + RND	Accuracy	0.555	0.555	0.555	0.555
Bing_Liu	Accuracy	0.635	0.635	0.635	0.635
Bing_Liu + RND	Accuracy	0.565	0.565	0.565	0.565
IMDB	Accuracy	0.6425	0.6425	0.6425	0.6425
IMDB + RND	Accuracy	0.5975	0.5975	0.5975	0.5975
8K	Accuracy	0.5025	0.5025	0.5025	0.5025
8K + RND	Accuracy	0.51	0.51	0.51	0.51
JST	Accuracy	0.7575	0.6375	0.7175	0.6325
	Perplexity	27.3111	26.9145	26.2463	25.2787
	Topic_Coh	-1.0123	-0.4089	-1.2434	-1.226
TJST	Accuracy	0.915	0.9425	0.9475	0.9325
	Perplexity	27.791	27.4122	26.993	26.5813
	Topic_Coh	-0.111	-1.6632	-0.199	-0.8503
RJST	Accuracy	0.62	0.54	0.5175	0.5175
	Perplexity	27.6284	26.5172	26.1489	25.1684
	Topic_Coh	-2.6713	-1.0947	-1.979	-2.3544
TS	Accuracy	0.5425	0.515	0.5175	0.5175
	Perplexity	27.6707	26.4786	26.1138	25.4488
	Topic_Coh	-1.5025	-1.3799	-1.0214	-4.6598
WJST	Accuracy1	0.7225	0.8475	0.7525	0.6975
	Accuracy2	0.71	0.8325	0.7575	0.6875
	Perplexity	23.1145	22.4588	21.2982	21.0494
WJST1	Topic_Coh	-0.0751	-0.9342	-0.0602	-0.4873
	Accuracy1	0.97	0.9575	0.9675	0.9675
	Accuracy2	0.9625	0.945	0.9675	0.96
WJST1	Perplexity	24.6302	23.5662	22.7180	22.1082
	Topic_Coh	-0.1348	-1.2359	-0.1329	-0.7102

TABLE 13: Sentiment classification on different datasets according to the different number of distinct windows (before random selection).

		Android					
Model	Metric\window	1	2	3	4	5	6
WJST	Accuracy1	0.8375	0.78	0.7925	0.7425	0.725	0.9075
	Accuracy2	0.83	0.77	0.7775	0.735	0.725	0.9075
	Perplexity	18.3948	16.7266	16.7303	16.166	15.9499	15.1128
	Topic_Coh	-1.4063	-0.9279	-0.5547	-1.2929	-1.7475	-0.746
WJST1	Accuracy1	0.8975	0.8525	0.81	0.8675	0.755	0.8025
	Accuracy2	0.8825	0.83	0.7925	0.8675	0.75	0.7875
	Perplexity	19.3734	18.0031	16.6787	16.8518	16.1015	16.5461
	Topic_Coh	-2.2084	-1.6272	-0.1870	-0.7753	-1.328	-1.7836
<i>Automotive</i>							
WJST	Accuracy1	0.7775	0.74	0.755	0.735	0.7375	0.69
	Accuracy2	0.7725	0.745	0.7475	0.745	0.725	0.685
	Perplexity	23.6365	21.7712	21.2008	20.9688	20.5095	19.6748
	Topic_Coh	-1.1684	-1.7095	-1.6542	-0.2824	-1.7604	-0.1311
WJST1	Accuracy1	0.805	0.8125	0.8	0.7575	0.78	0.7725
	Accuracy2	0.805	0.7975	0.79	0.755	0.78	0.7725
	Perplexity	23.2684	22.2601	21.4357	20.8092	20.5091	20.2379
	Topic_Coh	-1.7318	-0.5912	-1.1883	-0.7637	-0.62	-0.5134
<i>Electronic</i>							
WJST	Accuracy1	0.845	0.7675	0.8625	0.7325	0.7775	0.7325
	Accuracy2	0.845	0.7575	0.875	0.7275	0.7825	0.7225
	Perplexity	22.561	22.0462	20.5489	20.7952	20.7495	20.0283
	Topic_Coh	-0.8471	-0.6207	-2.0442	-1.4345	-0.786	-0.9412
WJST1	Accuracy1	0.8025	0.875	0.79	0.8675	0.8625	0.835
	Accuracy2	0.7975	0.87	0.8	0.865	0.8625	0.8375
	Perplexity	23.5903	23.546	22.6012	22.8559	21.8897	21.4464
	Topic_Coh	-0.8251	-0.9581	-1.1683	-1.2189	-0.8492	-0.3402
<i>Movie</i>							
WJST	Accuracy1	0.855	0.815	0.7225	0.6575	0.7	0.765
	Accuracy2	0.845	0.7975	0.71	0.6625	0.685	0.765
	Perplexity	24.5779	24.5153	23.1145	23.1209	23.2709	22.2328
	Topic_Coh	-0.4063	-0.0216	-0.0751	-2.5351	-0.0888	-0.0791
WJST1	Accuracy1	0.9725	0.9775	0.97	0.965	0.575	0.595
	Accuracy2	0.96	0.965	0.9625	0.955	0.5875	0.5725
	Perplexity	26.117	25.09	24.6302	23.9778	22.7068	22.3572
	Topic_Coh	-0.1348	-0.1348	-0.1348	-0.0315	-0.0378	-0.0106
<i>Average section</i>							
WJST	Accuracy1	0.8287	0.7756	0.7831	0.7168	0.735	0.7737
	Accuracy2	0.8231	0.7675	0.7775	0.7175	0.7293	0.77
	Perplexity	22.2925	21.2648	20.3986	20.2627	20.1199	19.2621
	Topic_Coh	-0.957	-0.8199	-1.082	-1.3862	-1.0956	-0.4743
WJST1	Accuracy1	0.8693	0.8793	0.8425	0.8643	0.7431	0.7512
	Accuracy2	0.8612	0.8656	0.8362	0.8606	0.745	0.7425
	Perplexity	23.0872	22.2248	21.3364	21.1236	20.3017	20.1469
	Topic_Coh	-1.225	-0.8278	-0.6696	-0.6973	-0.7087	-0.6619

4.2. *Topic Discovery.* The topics are extracted from datasets based on  $P(w|z)$  in this section. A topic is a multinomial distribution over words based on topics, sentiments, weights, and window sizes. The top words could approximately reflect the meaning of a topic. Tables 17–19 show some examples of topics extracted from Movie, Android, and Electronic datasets by different models. Each row shows the top 10 words for the corresponding topic and sentiment label. The top 10 words from each topic were extracted and then used for topic\_coherency. Extracting topic words is one of the main tasks in topic modeling. This section lists the top 10 words in three examples for Movie, Android, and Electronic datasets. The listed words for each topic describe

the topic. The listed words for the proposed methods have a better topic\_coherency value than baseline methods because they have a higher value of topic\_coherency. The higher the topic\_coherency value, the better the quality of the subjects extracted.

4.3. *Sentiment Classification at Document-Level.* In this section, the number of distinct windows is three, and the models use the AFINN sentiment dictionary in the initialization section of the Gibbs sampling algorithm. A document is classified based on  $P(s|\mathbf{r})$ , which is the probability of a sentiment given by a document. A document

TABLE 14: Sentiment classification on different datasets according to the different number of distinct windows (after random selection).

		Android					
Model	Metric\window	1	2	3	4	5	6
WJST	Accuracy1	0.71	0.775	0.7025	0.69	0.7175	0.675
	Accuracy2	0.7	0.765	0.6975	0.68	0.705	0.6675
	Perplexity	17.791	15.8544	16.1124	15.6932	13.8904	14.1876
	Topic_Coh	-1.6	-1.9691	-1.4526	-4.6888	-2.6353	-1.2267
WJST1	Accuracy1	0.8025	0.85	0.845	0.8025	0.8675	0.76
	Accuracy2	0.7925	0.83	0.84	0.7875	0.865	0.7525
	Perplexity	18.9319	17.7849	17.6145	15.8017	15.4004	15.0917
	Topic_Coh	-1.2666	-1.6744	-3.027	-2.3428	-1.1215	-2.0183
<i>Automotive</i>							
WJST	Accuracy1	0.7425	0.7375	0.68	0.6825	0.6725	0.64
	Accuracy2	0.74	0.73	0.6775	0.68	0.6625	0.6325
	Perplexity	20.8433	20.7212	19.6843	20.0801	19.1691	18.7129
	Topic_Coh	-1.4406	-2.0489	-2.6701	-2.2155	-1.1019	-1.1651
WJST1	Accuracy1	0.7825	0.76	0.755	0.7725	0.7475	0.7675
	Accuracy2	0.785	0.7575	0.7475	0.775	0.7525	0.76
	Perplexity	22.1856	21.3601	20.6057	19.8915	19.402	19.0827
	Topic_Coh	-1.5638	-1.474	-1.1274	-1.7761	-1.2042	-1.1755
<i>Electronic</i>							
WJST	Accuracy1	0.7975	0.72	0.76	0.6875	0.7675	0.6825
	Accuracy2	0.795	0.715	0.7475	0.66	0.765	0.665
	Perplexity	21.9171	21.0509	20.4858	20.26	19.4089	19.5549
	Topic_Coh	-0.8282	-1.0407	-0.9593	-0.9779	-1.0413	-0.6259
WJST1	Accuracy1	0.7475	0.7425	0.7775	0.7425	0.7725	0.78
	Accuracy2	0.745	0.7475	0.78	0.74	0.7625	0.785
	Perplexity	23.2269	22.4573	21.7203	21.4598	20.8683	20.5084
	Topic_Coh	-1.7905	-1.5082	-1.4854	-0.9143	-1.2881	-1.3489
<i>Movie</i>							
WJST	Accuracy1	0.7425	0.77	0.8075	0.615	0.7	0.595
	Accuracy2	0.7425	0.765	0.7875	0.59	0.69	0.595
	Perplexity	23.7529	23.322	22.1056	21.8573	20.8467	20.9653
	Topic_Coh	-1.0908	-0.9134	-0.9145	-0.6129	-1.1099	-0.716
WJST1	Accuracy1	0.9725	0.965	0.96	0.9675	0.96	0.9625
	Accuracy2	0.975	0.96	0.955	0.955	0.9575	0.9575
	Perplexity	26.1797	25.3138	24.5149	24.1116	23.5025	23.3023
	Topic_Coh	-0.0218	-0.2225	-1.9033	-0.0897	-0.2896	-0.0569
<i>Average section</i>							
WJST	Accuracy1	0.7481	0.7506	0.7375	0.6687	0.7143	0.6481
	Accuracy2	0.7443	0.7437	0.7275	0.6525	0.7056	0.64
	Perplexity	21.076	20.2371	19.597	19.4726	18.3287	18.3551
	Topic_Coh	-1.2399	-1.493	-1.4991	-2.1237	-1.4721	-0.9334
WJST1	Accuracy1	0.8262	0.8293	0.8343	0.8212	0.8368	0.8175
	Accuracy2	0.8243	0.8237	0.8306	0.8143	0.8343	0.8137
	Perplexity	22.631	21.729	21.1138	20.3161	19.7933	19.4962
	Topic_Coh	-1.1606	-1.2197	-1.8857	-1.2807	-0.9758	-1.1499

is classified as negative if  $P(+|\mathbf{r}) < P(-|\mathbf{r})$  and vice versa. Determining sentiment is important which is calculated using two formulas in this paper. In the first formula,  $P(\mathbf{s}|\mathbf{r}) = N_{s,r}/N_r$  where  $N_{s,r}$  is the number of words with sentiment  $s$  in document  $r$  and  $N_r$  is the number of words in document  $r$ . In the second formula,  $P(\mathbf{s}|\mathbf{r}) = F_{s,r}/F_r$  where  $F_{s,r}$  is the effect of words with sentiment  $s$  in document  $r$  and  $F_r$  is equal to the sum of the effect of words with different sentiments in document  $r$ . In all evaluations, accuracy1 is calculated based on the first formula, and accuracy2 is calculated based on the second formula. As shown in Figure 8, the document is negative according to the first

formula, and the document is positive according to the second formula, and the weight of positive words is more than negative ones, although the number of negative words is more than positive ones, and positive words can affect sentiment analysis at document-level.

In this section, the best values for each method (the highest accuracy, the lowest perplexity, and the highest topic\_coherency) are selected from Tables 9–12 and are listed in Tables 6 and 7. Table 6 compares the models based on four datasets (Android, Automotive, Movie, and Electronic) and Table 7 compares the models based on six datasets (Magazine, Sports, MR, Amazon, IMDB, and Yelp).

TABLE 15: Sentiment scores, for some instance, words related to Android, Automotive, Electronic, and Movie datasets.

Dataset	Android		Automotive		Electronic		Movie	
Model	Word	Score	Word	Score	Word	Score	Word	Score
WJST1	Nice	5	Much	5	Satisfy	5	See	5
	Cute	4	Use	4	Crew	4	Father	4
	Favorit	3	Long	3	Way	3	Pray	3
	Perfect	2	Expens	2	Fluid	2	Human	2
	Great	1	Stuff	1	Feel	1	Event	1
	Type	-1	Fals	-1	Pull	-1	Terribl	-1
	Wast	-2	Serious	-2	Side	-2	Sens	-2
	Everi	-3	Extens	-3	Nervous	-3	Lost	-3
	Unknown	-4	Space	-4	Even	-4	Sure	-4
Everyth	-5	Know	-5	Extend	-5	Injur	-5	
WJST	Nice	4	Much	3	Satisfy	-2	See	-4
	Cute	1	Use	2	Crew	2	Father	5
	Favorit	2	Long	4	Way	4	Pray	-3
	Perfect	2	Expens	-4	Fluid	1	Human	5
	Great	2	Stuff	-5	Feel	-3	Event	1
	Type	1	Fals	-5	Pull	-4	Terribl	4
	Wast	-5	Serious	1	Side	-1	Sens	-4
	Everi	4	Extens	5	Nervous	3	Lost	3
	Unknown	5	Space	-5	Even	-2	Sure	2
Everyth	2	Know	4	Extend	-5	Injur	-5	

TABLE 16: Sentiment scores for some instance words related to STS, Sanders, and SOMD datasets.

Model	Dataset	Word	STS Score	Sanders Score	SOMD Score
WJST1		Much	-4	4	2
		Good	5	4	-5
		Bad	-5	-1	-5
		Nice	-3	5	5
		Hate	-5	-5	1
		Love	5	5	3
WJST		Much	-5	-5	-2
		Good	-4	-1	3
		Bad	-4	-1	3
		Nice	3	2	1
		Hate	-5	-4	2
		Love	4	-2	-3

TABLE 17: Top 10 words extracted from the Movie dataset.

Model	Sentiment	Top 10 words
WJST	+	jesu, film, God, love, mel, Christian, life, suffer, believ, roman
	-	movi, godzilla, bad, dvd, origin, horror, buy, version, worst, actor
WJST1	+	Jesu, mel, passion, mother, stori, realli, great, everyon, God, like
	-	godzilla, monster, go, time, star, know, kill, make, militari, American
JST	+	mel, stori, mother, two, realli, becom, anoth, God, like, back
	-	godzilla, look, monster, american, militari, like, worst, zellweg, emmerich, quit

The results of Tables 6 and 7 are evaluated on unigram words. AFINN method classifies each document according to the  $P(s|r) = N_{s,r}/N_r$  where the word sentiment label is directly obtained from the AFINN sentiment lexicon. The RND method classifies each document according to the  $P(s|r) = N_{s,r}/N_r$  where the word sentiment label is

determined randomly, and in the AFINN + RND method, the algorithm uses both AFINN and RND methods. The improvement over these methods will reflect how much the proposed methods and baseline methods can learn from a dataset. The report in Tables 6 and 7 shows that the proposed models perform better than JST. Based on the results, the



TABLE 18: Top 10 words extracted from Android dataset.

Model	Sentiment	Top 10 words
WJST	+	app, game, sudoku, play, version, enjoy, option, want, hint, like
	-	work, app, would, fire, live, station, tri, say, select, kindl, load, user
WJST1	+	sudoku, tri, love, game, time, easi, tablet, star, call, make
	-	close, tablet, seem, get, year, download, much, station, time, android
JST	+	station, want, even, peopl, work, avail, version, puzzl, custom, believ
	-	use, app, find, review, great, got, total, new, night, fake

TABLE 19: Top 10 words extracted from Electronic dataset.

Model	Sentiment	Top 10 words
WJST	+	read, book, screen, touch, kindl, page, better, wifi, ebook, like
	-	work, went, new, servic, need, bad, system, hous, number, mine
WJST1	+	googl, amazon, book, color, store, kindl, download, small, pdf
	-	time, work, two, much, one, power, comput, phone, go, unit
JST	+	book, touch, read, page, free, librari, touch, screen, much, pdf
	-	plug, work, could, devic, comput, charger, router, cabl, item, design

proposed methods have a significant improvement over AFINN and the baseline methods on all datasets. As seen from AFINN-based methods results, the results calculated based on the sentiment lexicon are below 70% for most datasets. In this study, parameters perplexity and topic\_coherency are not calculated for AFINN, RND, and AFINN + RND methods. TS and RJST methods have lower accuracy than other methods on all datasets, but JST and TJST achieve better performance. As can be seen from the results, TJST outperforms JST on all datasets because, in JST, the distribution  $\theta$  depends on the document, but in TJST, the distribution  $\theta$  does not depend on the document and is generally estimated because it uses all documents for computations. According to Tables 6 and 7, WJST1 has higher accuracy than other methods. WJST1 outperforms WJST because, in WJST, the distributions  $\theta$ ,  $\xi$ , and  $\psi$  depend on the document, but in WJST1, the distributions  $\theta$ ,  $\xi$ , and  $\psi$  do not depend on the document and are generally estimated because they use all documents for computations. The perplexity value varies on different datasets because the size of datasets is different, according to Table 4.

The analysis of the Friedman test on the results of Tables 6 and 7 demonstrates that there is a statistically significant difference between the performances of the algorithms in terms of accuracy with  $\chi^2(10) = 92.091$  and  $p < 0.01$ , in terms of perplexity with  $\chi^2(5) = 38.629$  and  $p < 0.01$ , and in terms of topic\_coherency with  $\chi^2(5) = 5.508$  and  $p > 0.1$ . The mean rank of the algorithms based on the Friedman test, which is demonstrated in Figure 9, indicates that WJST1 ranks first among all the algorithms in 7 in terms of accuracy and topic\_coherency. According to Figure 9, if the experiment intends to find the minimum value (perplexity), the Friedman test assigns the lowest rank to the best-performing algorithm. If the problem intends to find the maximum value (accuracy and topic\_coherency), the Friedman test assigns the highest rank to the best-performing algorithm.

According to Figure 9, -1 is accuracy1 and -2 is accuracy2. As shown in Figure 10, average values of accuracy, perplexity, and topic\_coherency are equal to the average values in each column of Tables 6 and 7 for each method, in which the values are calculated on Android, Automotive, Electronic, Movie, Magazine, Sport, MR, Amazon, IMDB, and Yelp datasets. According to the results, WJST has a lower perplexity value than other methods. WJST1 outperforms WJST and baseline methods in terms of accuracy and topic\_coherency. According to Figure 10, -1 is accuracy1 and -2 is accuracy2.

*4.4. Evaluation Results According to the Different Situations, with AFINN and NO\_AFINN States.* In this section, the study aims to examine the impact of the AFINN dictionary in the initialization part of Gibbs sampling on the proposed models. The results of the evaluation are shown in Table 8. In this section, the number of distinct windows is three, and the number of topics is ten. The most effective is visible in WJST1 on the Movie dataset, where the accuracy in the NO\_AFINN state is equal to 0.48 and is equal to 0.95 in the AFINN state. Prior sentiment information affects perplexity and topic\_coherency lower than accuracy. According to Table 8, it can be seen that using the AFINN dictionary is more effective than using the NO\_AFINN state. In the NO\_AFINN state, prior sentiment information was not incorporated into the models for sentiment words in the initialization section of the Gibbs sampling algorithm.

*4.5. Evaluation Results According to the Different Sentiment Dictionaries.* In this subsection, the study compares different dictionaries achieved by the proposed models. The output of WJST and WJST1 can be a weighted sentiment dictionary. Using the obtained dictionary by each method on each dataset, other datasets will be evaluated. Each document will be classified according to  $P(s|\mathbf{r})$ , where the word

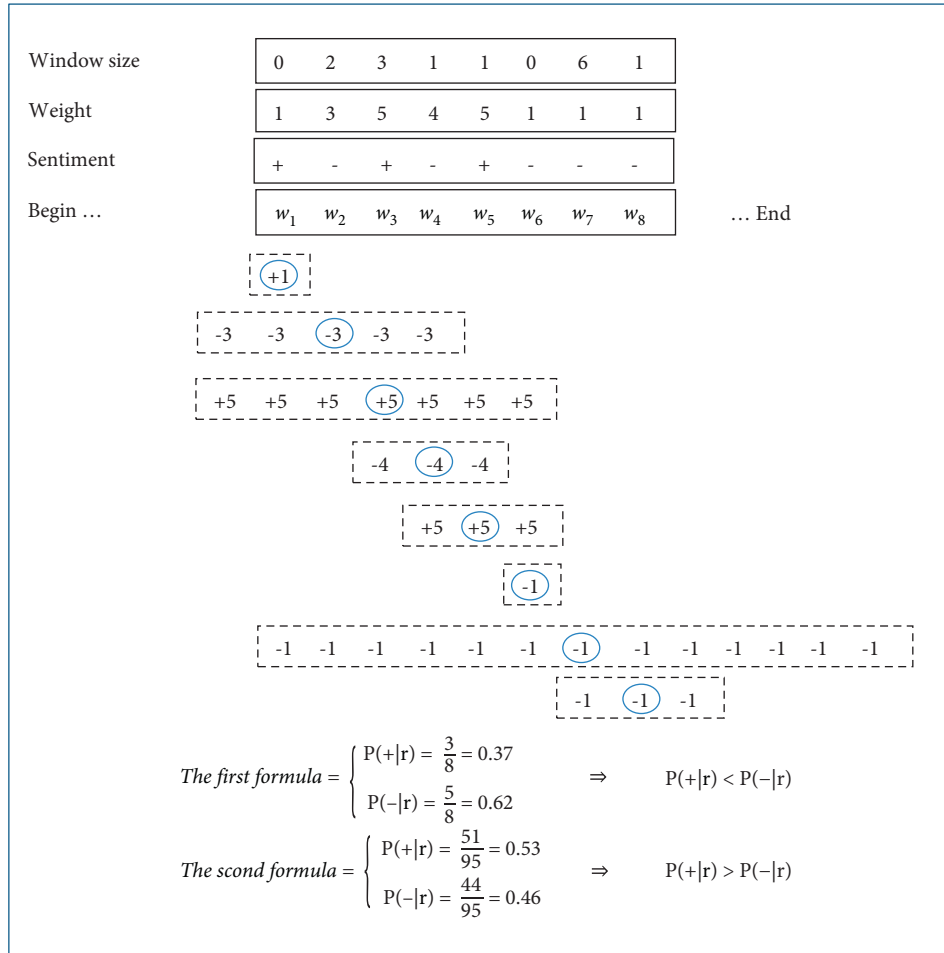


FIGURE 8: An example of calculating the sentiment of a document using two formulas.

sentiment label is directly obtained from the dictionary. Tables 20 and 21 are related to WJST and WJST1, respectively. The impact of using different dictionaries achieved by WJST and WJST1 is presented in Tables 20 and 21. The methods AFINN +  $w$ , Android +  $w$ , ELEC +  $w$ , Auto +  $w$ , and MOV +  $w$  classify each document according to  $P(s|r) = F_{s,r}/F_r$ , where the weight and sentiment label is directly obtained from AFINN, Android, Electronic, Automotive, and Movie lexicons, and window size is considered to be one for all words in all documents. Methods Bing\_Liu, 8K, Android, Automotive, ELEC, MOV, and IMDB classify each document according to  $P(s|r) = N_{s,r}/N_r$  where the word sentiment label is directly obtained from Bing\_Liu, 8K, Android, Automotive, Electronic, Movie, and IMDB lexicons, respectively. The Bing\_Liu + RND method uses both Bing\_Liu and RND methods. In the IMDB + RND method, the algorithm uses both IMDB and RND methods. The 8K + RND method utilizes both 8K and RND methods. In the Android + RND method, the algorithm uses both Android and RND methods. In the Auto + RND method, the algorithm uses both Auto and RND methods. In the ELEC + RND method, the algorithm uses both ELEC and RND methods. In the MOV + RND method, the algorithm uses both MOV and RND methods. According to Table 20,

the AFINN method achieves the highest accuracy on one dataset. Proposed methods achieve the highest accuracy on six datasets. According to the results in Table 21, the proposed methods achieve the highest accuracy on seven datasets. In AFINN, Bing Liu, IMDB, and 8-k dictionaries, sentiment and score values are set manually for each word, but proposed models use topic modeling to generate dictionaries. Proposed methods such as MOV and ELEC perform well on the datasets on which they are created based on. The dictionaries achieved by the proposed models are dependent on the application domain.

The analysis of the Friedman test on the results of Table 20 demonstrates that there is a statistically significant difference between the performances of competitors in terms of accuracy with  $\chi^2(27) = 70.070$  and  $p < 0.01$ . The analysis also shows that there is a statistically significant difference between the performances of the algorithms in Table 21 in terms of accuracy with  $\chi^2(27) = 79.740$  and  $p < 0.01$ . The mean rank of the algorithms can be seen in Figure 11. As shown in Figure 12, *Average1* is equal to the average of values in each row for each method, in which the values are calculated on datasets Android, Automotive, Electronic, Movie, STS, Sanders, and SOMD based on Tables 20 and 21. Furthermore, *Average2* is equal to the average of values in

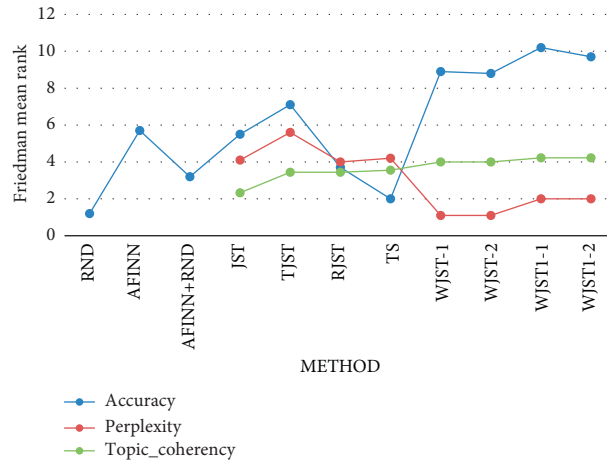
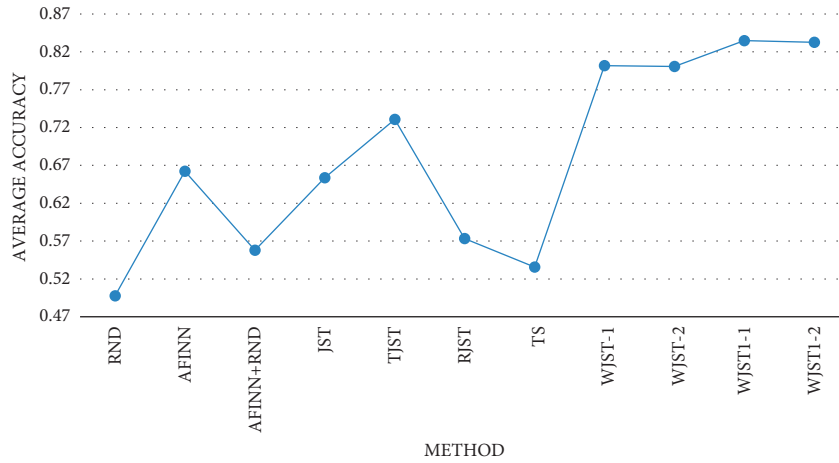
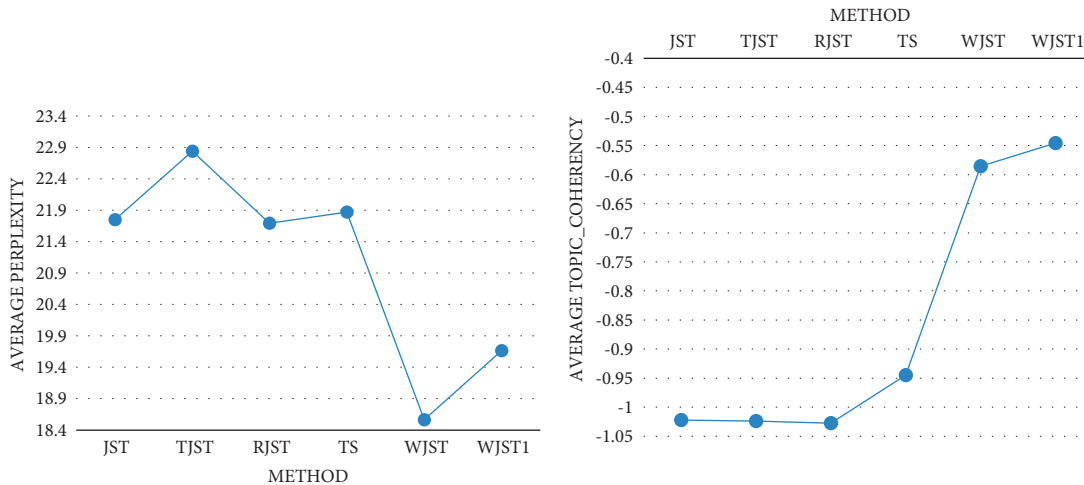


FIGURE 9: According to the Friedman test, the mean rank of algorithms in Tables 6 and 7.



(a)



(b)

(c)

FIGURE 10: Average of sentiment classification values calculated on Android, Automotive, Electronic, Movie, Magazine, Sport, MR, Amazon, IMDB, and Yelp datasets (based on Tables 6 and 7), in terms of accuracy (a), perplexity (b), and topic\_coherency (c).

each row for each method, in which the values are calculated on Android, Automotive, Electronic, and Movie based on Tables 20 and 21. According to the results, Bing\_Liu achieves

the highest value on column *Average1*. Furthermore, the Android, MOV, and Bing\_Liu methods have higher accuracy than other methods.

TABLE 20: Sentiment classification using different sentiment dictionaries achieved by WJST.

Model\dataset	Android	Auto	ELEC	MOV	STS	Sanders	SOMD
RND	0.48	0.4925	0.465	0.525	0.5346	0.4852	0.4847
AFINN	0.6975	0.625	0.675	0.595	0.734	0.674	0.3395
AFINN + w	0.685	0.58	0.6375	0.595	—	—	—
AFINN + RND	0.58	0.535	0.52	0.555	0.6038	0.5424	0.4475
Bing_Liu	0.6975	0.64	0.695	0.635	0.698	0.6813	0.322
Bing_Liu + RND	0.5775	0.535	0.535	0.565	0.6288	0.5416	0.4388
IMDB	0.7025	0.59	0.6375	0.6425	0.5512	0.5988	0.4617
IMDB + RND	0.6125	0.4825	0.5475	0.5975	0.5595	0.5196	0.4748
8K	0.5425	0.5025	0.5075	0.5025	0.4986	0.5351	0.3995
8K + RND	0.515	0.48	0.5075	0.51	0.5373	0.495	0.4814
Android	0.8525	0.62	0.645	0.6125	0.6023	0.6078	0.4712
Android + w	0.8375	0.59	0.63	0.62	—	—	—
Android + RND	0.8525	0.6275	0.6675	0.61	0.5995	0.5865	0.4832
Auto	0.57	0.705	0.6375	0.605	0.6023	0.5767	0.6011
Auto + w	0.5775	0.7	0.65	0.6075	—	—	—
Auto + RND	0.5625	0.705	0.6175	0.5875	0.6106	0.5522	0.588
ELEC	0.6525	0.5975	0.8425	0.4925	0.6023	0.6029	0.5323
ELEC + w	0.6625	0.605	0.8275	0.5025	—	—	—
ELEC + RND	0.645	0.6075	0.8425	0.4775	0.6023	0.6037	0.5585
MOV	0.66	0.5925	0.6375	0.81	0.6244	0.535	0.5127
MOV + w	0.6725	0.59	0.6475	0.8025	—	—	—
MOV + RND	0.6525	0.5875	0.62	0.81	0.6217	0.5416	0.5105
STS	0.51	0.58	0.575	0.615	0.624	0.4762	0.5312
STS + RND	0.5575	0.5625	0.5275	0.5775	0.624	0.5138	0.54
Sanders	0.59	0.6	0.565	0.5575	0.5607	0.7088	0.5105
Sanders + RND	0.5925	0.5925	0.5825	0.5425	0.5746	0.7088	0.5443
SOMD	0.5525	0.5275	0.47	0.4625	0.4859	0.5334	0.6093
SOMD + RND	0.5375	0.5	0.4875	0.4725	0.5441	0.5236	0.6093

TABLE 21: Sentiment classification using different sentiment dictionaries achieved by WJST1.

Model\dataset	Android	Auto	ELEC	MOV	STS	Sanders	SOMD
RND	0.48	0.4925	0.465	0.525	0.5346	0.4852	0.4847
AFINN	0.6975	0.625	0.675	0.595	0.734	0.674	0.3395
AFINN + w	0.685	0.58	0.6375	0.595	—	—	—
AFINN + RND	0.58	0.535	0.52	0.555	0.6038	0.5424	0.4475
Bing_Liu	0.6975	0.64	0.695	0.635	0.698	0.6813	0.322
Bing_Liu + RND	0.5775	0.535	0.535	0.565	0.6288	0.5416	0.4388
IMDB	0.7025	0.59	0.6375	0.6425	0.5512	0.5988	0.4617
IMDB + RND	0.6125	0.4825	0.5475	0.5975	0.5595	0.5196	0.4748
8K	0.5425	0.5025	0.5075	0.5025	0.4986	0.5351	0.3995
8K + RND	0.515	0.48	0.5075	0.51	0.5373	0.495	0.4814
Android	0.855	0.6025	0.565	0.65	0.5857	0.5865	0.4985
Android + w	0.835	0.6	0.6025	0.6625	—	—	—
Android + RND	0.855	0.59	0.575	0.6125	0.5967	0.5579	0.505
Auto	0.6375	0.745	0.6175	0.63	0.594	0.6233	0.4766
Auto + w	0.6325	0.7375	0.65	0.635	—	—	—
Auto + RND	0.6375	0.745	0.62	0.6225	0.5829	0.6118	0.4875
ELEC	0.65	0.5475	0.7075	0.5825	0.594	0.6192	0.457
ELEC + w	0.63	0.5775	0.7125	0.63	—	—	—
ELEC + RND	0.6425	0.5575	0.7075	0.5775	0.6134	0.5914	0.4744
MOV	0.63	0.5525	0.5825	0.6375	0.5663	0.5767	0.4603
MOV + w	0.6425	0.5475	0.6425	0.94	—	—	—
MOV + RND	0.635	0.565	0.61	0.6375	0.5718	0.571	0.4493
STS	0.585	0.5475	0.595	0.5775	0.7431	0.5939	0.4231
STS + RND	0.5875	0.565	0.5975	0.5725	0.7431	0.5743	0.4395
Sanders	0.605	0.555	0.6	0.5375	0.6632	0.7538	0.421
Sanders + RND	0.6075	0.55	0.6025	0.5375	0.6771	0.7538	0.4559
SOMD	0.615	0.555	0.55	0.5125	0.5773	0.6078	0.594
SOMD + RND	0.61	0.5275	0.5425	0.5325	0.594	0.5767	0.594

**4.6. Evaluation Results According to the Different Number of Topics.** In this subsection, the proposed models are examined based on the different topics (5, 10, 15, and 20). The AFINN dictionary is utilized in methods, and the number of distinct windows is three. Evaluations results are shown in Tables 9–12. The proposed methods are better than the baseline methods based on the results. The results show that increasing the number of topics will decrease the perplexity value. WJST1 achieves the highest accuracy on the Movie dataset with 97 percent, but the highest accuracy value on the Movie dataset in WJST is equal to 84 percent. Results show that the proposed methods perform better with different topic number settings, especially for WJST1 with 97% accuracy at  $|Z|=5$  on the Movie dataset. Based on the results, WJST has a lower perplexity than other methods. WJST1 outperforms WJST and baseline methods in terms of accuracy and topic\_coherency. TJST performs better than the WJST method in terms of accuracy, but WJST achieves higher accuracy than JST and other baseline methods. This observation shows that modeling the parameters *weight* and *window* improves sentiment classification at the document-level. According to (18), a lower topic\_coherency value suggests that the retrieved subjects are of worse quality than one with a highertopic\_coherency. The words in a subject accurately describe the subject and have a stronger association with one another.

**4.7. Evaluations Results According to the Different Number of Distinct Windows.** In this subsection, the proposed models are evaluated according to the different number of separate windows (1, 2, 3, 4, 5, and 6), which are effective for improving the proposed models. In this experiment, the number of topics is five, and the models use the AFINN sentiment dictionary. Based on Table 13, the proposed methods are compared according to accuracy, perplexity, and topic\_coherency. The results show that increasing the number of distinct windows will decrease the perplexity value. In the report in Table 13, an increase in the size of a window will reduce the accuracy because it will increase the number of words in the window, and each term may not affect all neighbors in its window.

For instance, as shown in Figure 13, the word *terrible* has a window size equal to 3. In Table 13, it is assumed that each word affects all neighbors in its window, so Table 13 takes that the unigram *terrible* effects unigrams *film*, *last*, *season*, *sophie*, *best*, and *actress*. As shown in Figure 13, the word *terrible* can affect unigrams *film*, *last*, and *season*, but it is not about unigrams *sophie*, *best*, and *actress*. So, finding the words that can be involved in a window is a new challenge that we introduce in this study, and two methods are presented. The first method assumes that each word affects all neighbors in its window, and the second method assumes that each word affects some random neighbors in its window. So, the first method selects all neighbors, but the second method selects some neighbors randomly. In this study, all evaluations are calculated based on the first method, and the second method is considered for the evaluation in Table 14.

As shown in Table 13, accuracy, perplexity, and topic\_coherency values are calculated on Android, Automotive, Electronic, and Movie datasets before random selection using the first method. As shown in Table 14, accuracy, perplexity, and topic\_coherency values are calculated on Android, Automotive, Electronic, and Movie datasets after random selection using the second method. So, in Table 13, it is assumed that each word affects all neighbors in its window, but in Table 14, it is assumed that each word affects some random neighbors in its window. As shown in Tables 13 and 14, average values of accuracy, perplexity, and topic\_coherency are equal to average values in each column of Tables 13 and 14 for each window size. The values are calculated on Android Automotive, Electronic, and Movie datasets. According to the results, the second method is more stable than the first method in terms of accuracy, but the first method has higher accuracy than the second method. The second method outperforms the first method in terms of perplexity. The first method performs better than the second in terms of accuracy and topic\_coherency.

The analysis of the Friedman test on the results of Table 13 demonstrates that there is a statistically significant difference between the performances of the algorithms in terms of accuracy with  $\chi^2(5) = 27.608$  and  $p < 0.01$ , in terms of perplexity with  $\chi^2(5) = 35.143$  and  $p < 0.01$ , and in terms of topic\_coherency with  $\chi^2(5) = 6.232$  and  $p = 0.284$ . The mean rank of the algorithms based on the Friedman test, which is demonstrated in Figure 14, indicates that ( $w=1$ ) outperforms other windows in terms of accuracy. Still, it has lower perplexity and topic\_coherency values than other windows. ( $w=6$ ) outperforms other windows in perplexity and topic\_coherency, but it has a lower accuracy value than other windows. ( $w=3$ ) provides a special situation for proposed algorithms in which accuracy, topic\_coherency, and perplexity values are between the highest and lowest values ( $w=1, 2, 4, 5$ , and  $6$ ). So, in this study, all evaluations are calculated based on ( $w=3$ ). The mean rank of the algorithms based on Table 14, which is demonstrated in Figure 15, indicates that ( $w=1$ ) ranks first among all the algorithms in Table 14 in terms of accuracy. The analysis of the Friedman test indicates that there is a statistically significant difference in terms of accuracy with  $\chi^2(5) = 17.550$  and  $p < 0.01$ , in terms of perplexity with  $\chi^2(5) = 37.857$  and  $p < 0.01$ , and in terms of topic\_coherency with  $\chi^2(5) = 5.857$  and  $p = 0.320$ .

**4.8. Sentiment Classification Using Proposed Methods in Comparison to ALGA.** In this subsection, WJST and WJST1 are compared to ALGA [46]. Three datasets have been selected for evaluating the methods. Evaluations results are shown in Figure 16, which compares the results of models with each other according to accuracy, perplexity, and topic\_coherency metrics. In ALGA [46], several sentiment lexicons are created for a dataset during the training stage using a genetic algorithm. During the testing process, these dictionaries are employed. Every dictionary has some words and scores. Each chromosome is represented as a vector of sentiment words and their scores in the genetic algorithm

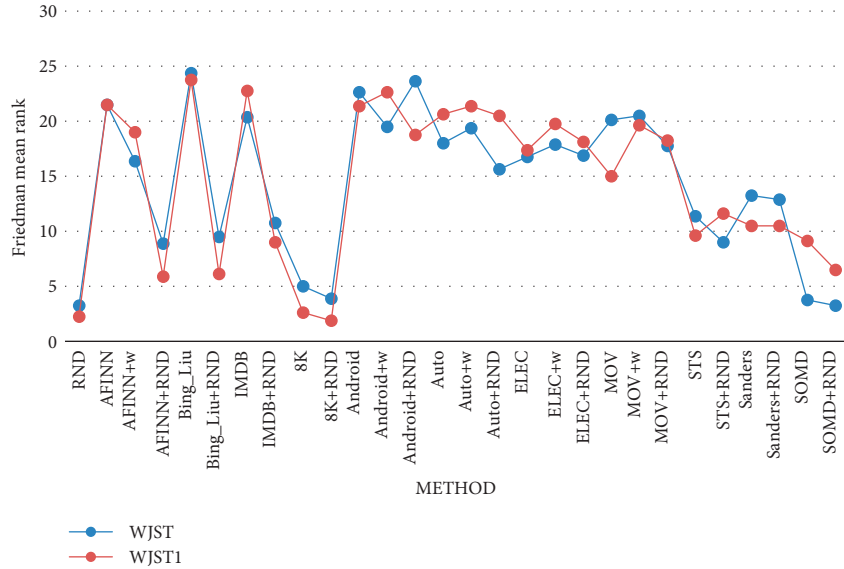


FIGURE 11: The mean rank according to the Friedman test based on results of Tables 20 and 21.

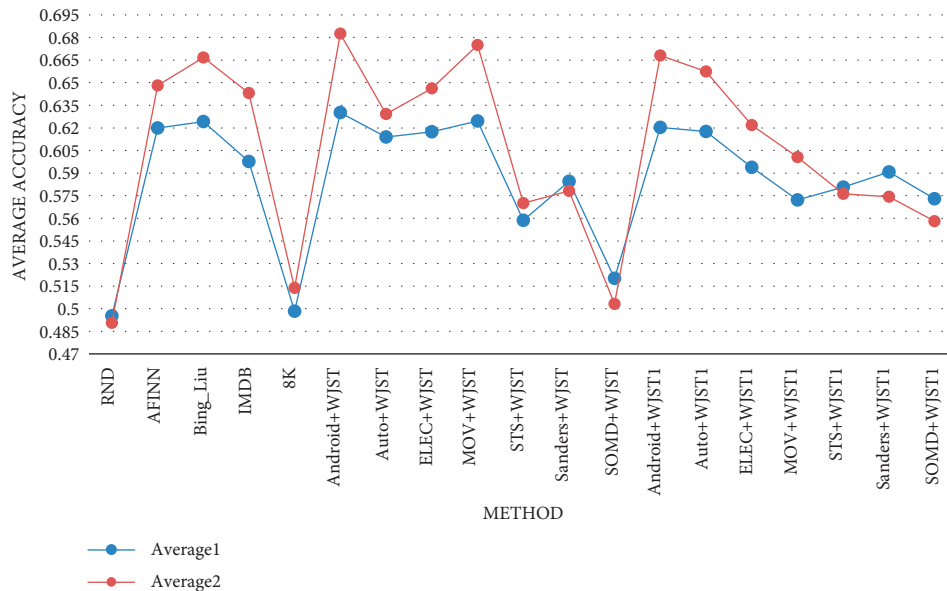


FIGURE 12: Average accuracy values are calculated using different sentiment dictionaries on Android, Automotive, Electronic, Movie, STS, Sanders, and SOMD datasets (based on Tables 20 and 21).

employed in the method. The scores are spread between a feeling word's lowest and maximum scores. The primary goal of ALGA is to create a dictionary that reduces errors on training datasets. The sum of scores for words of each instance  $T_i$  in dataset  $D_m$  using dictionary  $L_k$  is calculated using equation (19) and is treated as a feature [46]:

$$ALGA(D_m, T_i, L_k) = \sum_{W_j \in T_i} v_k(W_j). \quad (19)$$

Finding the values of words in the dictionaries (chromosomes) and adding them together is how the ALGA value for each instance is calculated. In (19),  $W_j$  represents the

words of  $T_i$ , and  $v_k(W_j)$  shows the score of  $W_j$  in  $L_k$ . As mentioned in [46], ALGA will predict a positive instance when the ALGA feature is positive and a negative instance when the ALGA feature is negative. By dividing the number of correct predictions of instances of a given dataset by the total cases, ALGA's accuracy is calculated. In this subsection, proposed methods are compared with ALGA [46] because it can automatically generate a sentiment dictionary. ALGA generates a sentiment dictionary using the genetic algorithm, but proposed methods generate a sentiment dictionary using topic modeling. In proposed models, each document is classified based on  $P(\mathbf{s} | \mathbf{r})$ , the probability of

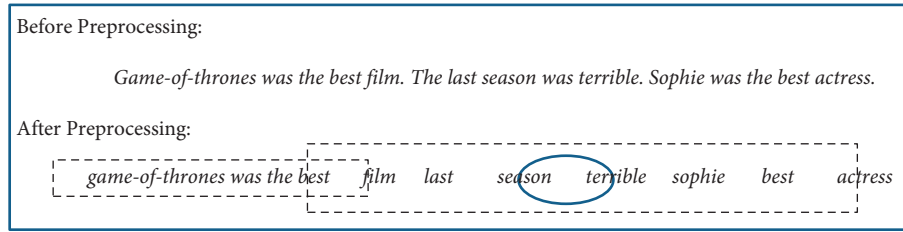


FIGURE 13: An example for showing the effect of each word on neighbors in its window.

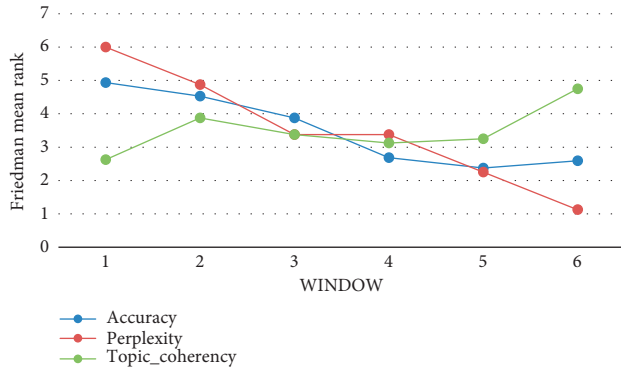


FIGURE 14: The mean rank of the seven algorithms is in Table 13 according to the Friedman test.

sentiment label given a document. In proposed models, two labels (+, -) are considered, and a document is classified as negative if  $P(+|\mathbf{r}) < P(-|\mathbf{r})$  and vice versa. Evaluations results can be seen in Figure 16. In this subsection, the number of distinct windows is three, and the number of topics is five. The models use the AFINN sentiment dictionary. According to Figures 16(a)–16(c), each column compares different methods on a dataset. The details of the datasets used in this section are illustrated in Table 4. In this subsection, only the accuracy is considered for the evaluation of ALGA. The ALGA-SW value is achieved by executing ALGA without taking stopwords into account. According to the results, WJST has higher accuracy than TJST on all datasets. WJST1 outperforms WJST and TJST on all datasets. ALGA and ALGA-SW perform better than other methods in terms of accuracy, but WJST1 achieves higher accuracy than ALGA and ALGA-SW on STS and Sanders datasets. The RND method achieves the lowest accuracy value on Sanders and STS datasets. The AFINN + RND method has higher accuracy than the RND method and has a lower accuracy than the AFINN method on Sanders and STS datasets. The RND method outperforms TJST, AFINN, and AFINN + RND methods on the SOMD dataset. In this study, parameters perplexity and topic\_coherency are not calculated for ALGA, ALGA-SW, AFINN, RND, and AFINN + RND methods. According to Figure 16, -1 is accuracy1, and -2 is accuracy2.

**4.9. Sentiment Classification Using Proposed Methods on Multidomain Datasets.** In this subsection, the performance of the proposed methods is compared with baseline methods

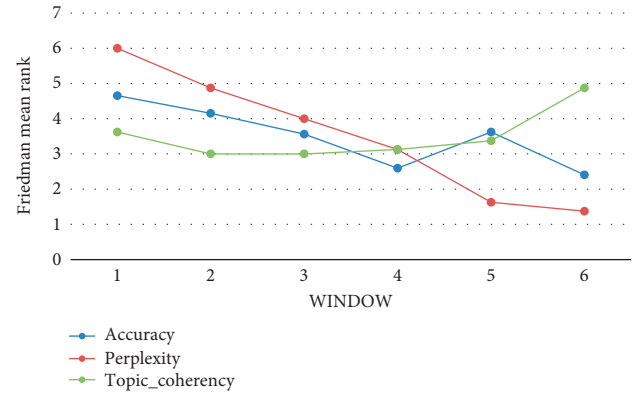


FIGURE 15: The mean rank of algorithms is in Table 14 according to the Friedman test.

on a multidomain dataset. In this experiment, the number of distinct windows and topics is three and five, respectively, and the models use the AFINN sentiment dictionary. The multidomain dataset contains reviews taken from multiple domains (product types). The details of the multidomain dataset used in this section are illustrated in Table 22.

As shown in Figure 17, accuracy, perplexity, and topic\_coherency values are calculated on a multidomain dataset that contains Android, Automotive, Electronic, and Movie domains. The methods WJST-dictionary and WJST1-dictionary classify each document according to  $P(s|\mathbf{r}) = N_{s,r}/N_r$ . The word sentiment label is directly obtained from WJST and WJST1 lexicons achieved by WJST and WJST1 on the multidomain dataset. According to Figure 17, -1 is accuracy1 and -2 is accuracy2. Based on the results, WJST has a lower perplexity value than other methods. WJST1 outperforms WJST in terms of topic\_coherency. WJST performs better than the WJST1 method in terms of accuracy because the distributions  $\theta$ ,  $\xi$ , and  $\psi$  in WJST depend on document, but in WJST1, the distributions  $\theta$ ,  $\xi$ , and  $\psi$  do not depend on the document. Dependency between documents of a domain is more than documents in different domains. A pattern in the documents of a domain may not exist in the documents of other domains. Therefore, calculations on multidomain datasets should be local and not cover all domains. For example, considering the distributions  $P(\mathbf{z}|\mathbf{s})$  and  $P(\mathbf{z}|\mathbf{s}, \mathbf{r})$ , where  $\mathbf{z}$  is topic,  $\mathbf{s}$  is the sentiment, and  $\mathbf{r}$  is documents. In the first state ( $P(\mathbf{z}|\mathbf{s})$ ), the topic depends on sentiment, and the distribution covers all documents in different domains. Perhaps a topic was positive in one domain and negative in another. Therefore, it is better to depend the topic on the documents of a domain, not all

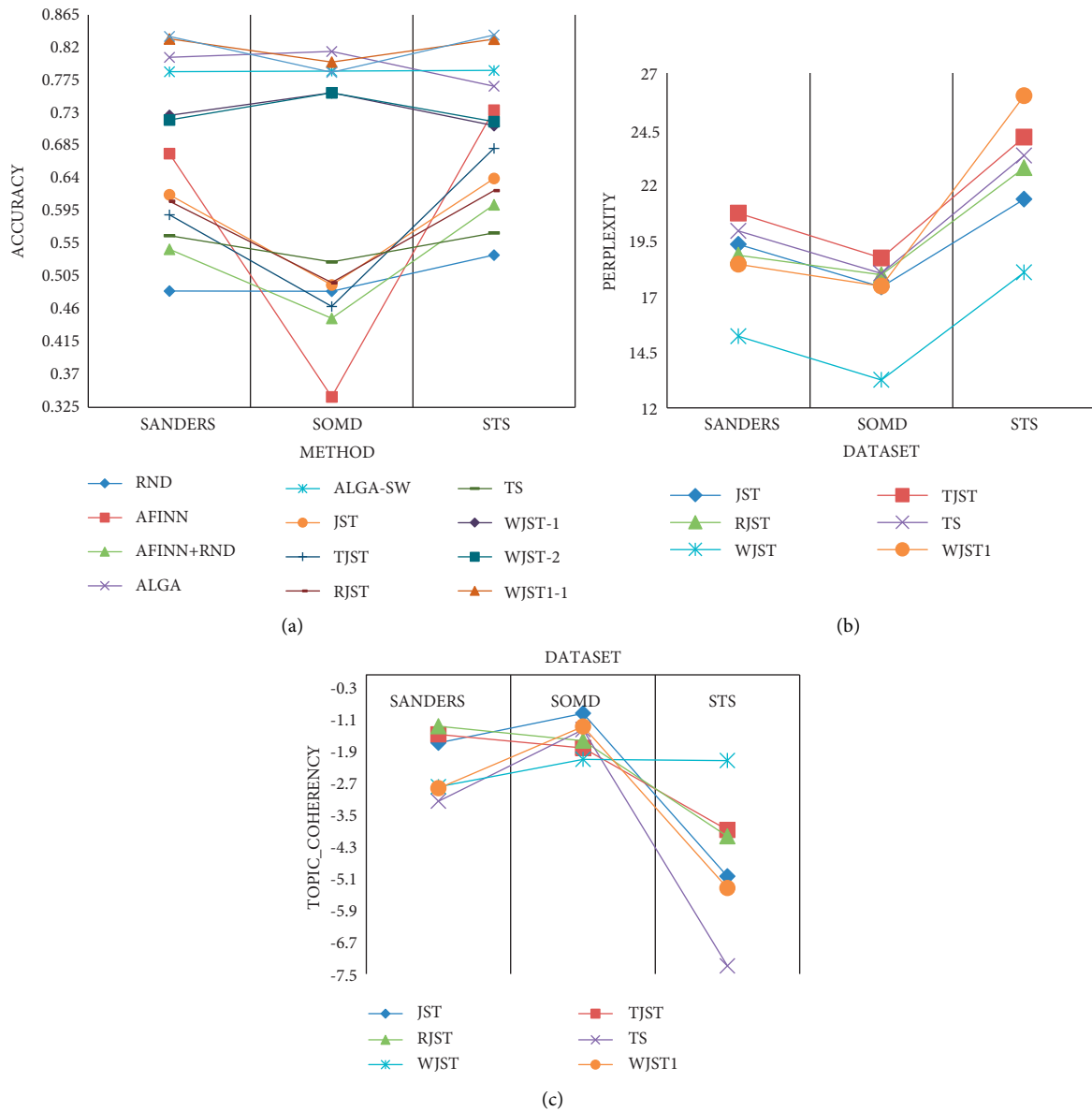


FIGURE 16: Sentiment classification on SANDERS, SOMD, and STS datasets, in terms of accuracy (a), perplexity (b), and topic\_coherency (c).

domains. Thus, the topic is limited to the document (and domain), and contradiction between different domains is eliminated. Therefore, WJST is suitable for multidomain datasets, and WJST1 is a version of WJST suitable for single-domain datasets. Sentiment classification on multidomain datasets is a challenge, and our solution in this study is using WJST, whose distributions ( $\theta$ ,  $\xi$ , and  $\psi$ ) depend on the document. Sentiment classification on multidomain datasets is a challenge, and further studies can be conducted to investigate this problem for future research.

**4.10. Comparison with Other Methods.** In this subsection, the best performance of the proposed methods is compared with 57 competitors [13, 76]; [82–88], [8–10, 46] which is shown

in Table 23. The details of the datasets used in this section are illustrated in Table 4.

**4.11. Comparison with Discriminative Models.** The proposed methods are compared to baseline approaches such as logistic regression and SVM on four datasets in the following experiment. The multidomain dataset contains Android, Automotive, Electronic, and Movie domains. As shown in Table 24, the accuracy value is calculated on four datasets. The results demonstrate that proposed methods have improved notably over AFINN and the baseline methods on all datasets. Based on Table 24, methods use two systems in preprocessing phase, which includes Bag of Word (BOW) and Term Frequency Inverse Document Frequency (TF-



TABLE 22: Description of the multidomain dataset used in this section.

#	Domains	Number of reviews	Vocabulary size	Number of words
1	Android, Automotive, Electronic, and Movie	1600	11183	100113

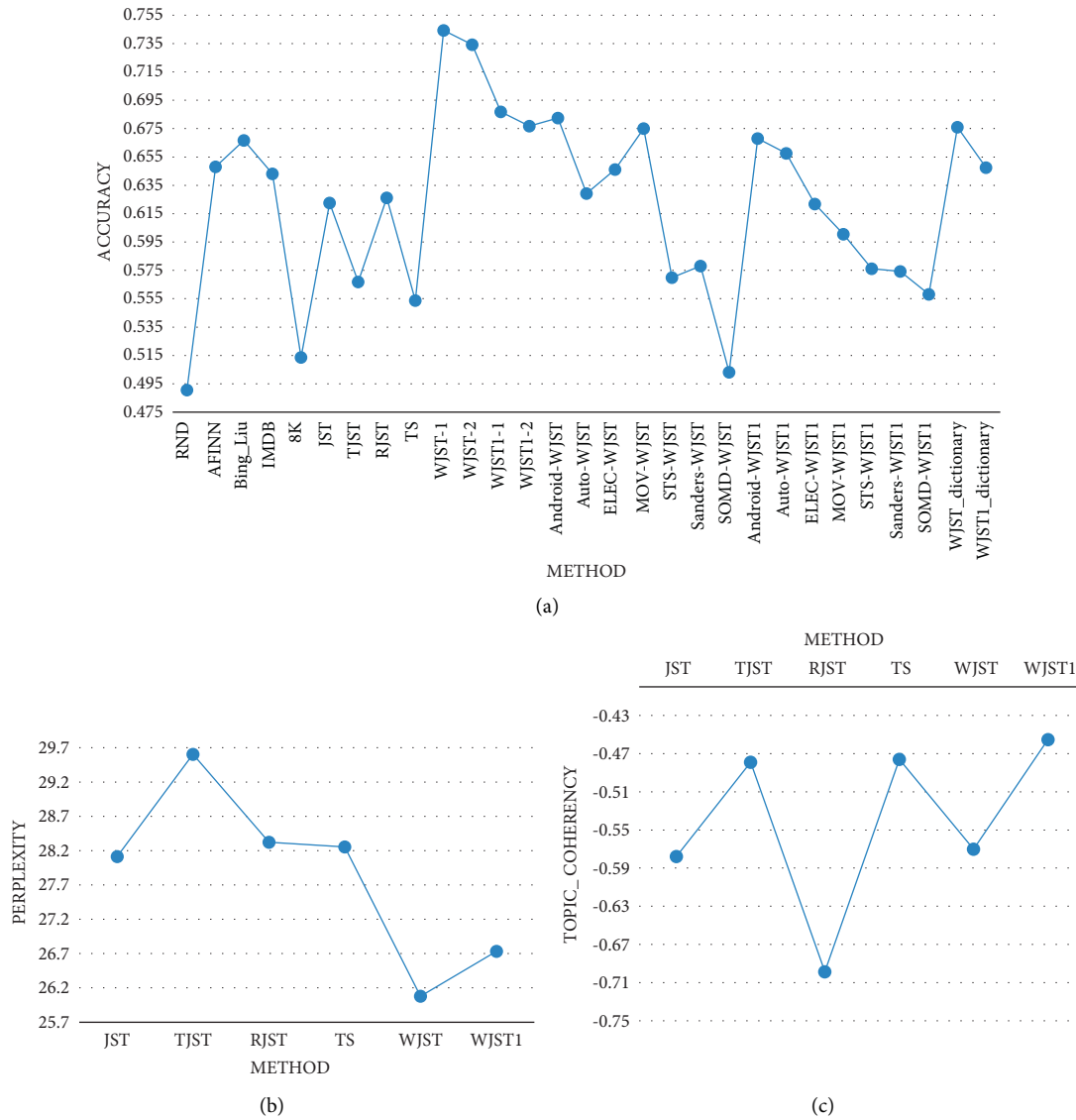


FIGURE 17: Sentiment classification values are calculated on a multidomain dataset in terms of accuracy (a), perplexity (b), and top-ic\_coherency (c).

IDF). In the BOW system, more word frequency reflects more importance of the word. TF-IDF system believes that high frequency may not be able to provide much information. Furthermore, rare words contribute more weight to the method. According to evaluation results, the results of the TF-IDF system are better than the BOW system.

4.12. Comparison with JST According to Extended Features. Suppose a unigram corresponds to the sentiment lexicon. In that case, its polarity will be equal to the subjectivity of the lexicon in order to identify the emotion label of the unigram for

trying to prepare prior emotion information. The following technique is used to decide the emotion label of a bigram to prepare prior emotion information: If words of the bigram have the same polarity, the bigram’s polarity will be the same as that of the words. If one of the words is in the lexicon, the bigram’s polarity will equal the lexicon’s subjectivity. The bigram’s polarity will be opposed to the polarity of the second word if the first word is ‘not.’ The following methodology is used to decide the emotion label of a trigram in order to prepare prior emotion information. If words of the trigram have the same polarity, the trigram’s polarity will be the same as that of the words. If one of them is in the lexicon, the trigram’s polarity will equal the

TABLE 23: Sentiment classification on MR, Sanders, SOMD, STS, Amazon, IMDB, and Yelp datasets.

Method/dataset	MR	Sanders	SOMD	STS	Amazon	IMDB	Yelp
ALGA [46]	—	0.8067	0.8147	0.7668	—	—	—
ALGA-SW [46]	—	0.7868	0.7877	0.7886	—	—	—
BPSO (Shang et al., 2016)	—	—	—	—	0.7439	0.79	0.789
BICA (Mirhosseini et al., 2017)	—	—	—	—	0.793	0.745	0.763
BABC (Schiezero et al., 2013)	—	—	—	—	0.7509	0.74	0.736
MaxEnt (Saif et al., 2014)	—	0.8362	—	0.7782	—	—	—
NB (Saif et al., 2014)	—	0.8266	—	0.8106	—	—	—
LS-all [13]	—	0.8199	—	—	—	—	—
SVM-all [13]	—	0.8214	—	—	—	—	—
RMTL [13]	—	0.827875	—	—	—	—	—
MTL-graph [13]	—	0.801725	—	—	—	—	—
CMSC [13]	—	0.846325	—	—	—	—	—
LSTM-all [13]	—	0.8063	—	—	—	—	—
MTL-CNN [13]	—	0.829825	—	—	—	—	—
MTL-DNN [13]	—	0.817	—	—	—	—	—
ASP-MTL [13]	—	0.85125	—	—	—	—	—
NeuroSent [13]	—	0.834575	—	—	—	—	—
DAM [13]	—	0.863225	—	—	—	—	—
SVM-BoW (Da Silva et al., 2014)	—	0.8243	0.7402	—	—	—	—
SVM-BoW + lex (Da Silva et al., 2014)	—	0.8398	0.7893	—	—	—	—
RF-BoW (Da Silva et al., 2014)	—	0.7924	0.7391	—	—	—	—
RF-BoW + lex (Da Silva et al., 2014)	—	0.8235	0.7936	—	—	—	—
LR-BoW (Da Silva et al., 2014)	—	0.7745	0.7238	—	—	—	—
LR-BoW + lex (Da Silva et al., 2014)	—	0.7949	0.7806	—	—	—	—
MNB-BoW (Da Silva et al., 2014)	—	0.7982	0.7543	—	—	—	—
MNB-BoW + lex (Da Silva et al., 2014)	—	0.8341	0.8013	—	—	—	—
ENS(LR + RF + MNB)-BoW (Da Silva et al., 2014)	—	0.8276	0.7555	—	—	—	—
ENS(LR + RF + MNB)-BoW + lex (Da Silva et al., 2014)	—	0.8489	0.8035	—	—	—	—
SVM-FH (Da Silva et al., 2014)	—	0.4975	0.5131	—	—	—	—
SVM-FH + lex (Da Silva et al., 2014)	—	0.7500	0.6299	—	—	—	—
RF-FH (Da Silva et al., 2014)	—	0.5564	0.6136	—	—	—	—
RF-FH + lex (Da Silva et al., 2014)	—	0.7163	0.7260	—	—	—	—
LR-FH (Da Silva et al., 2014)	—	0.5694	0.6529	—	—	—	—
LR-FH + lex (Da Silva et al., 2014)	—	0.7598	0.7303	—	—	—	—
MNB-FH (Da Silva et al., 2014)	—	0.5425	0.6070	—	—	—	—
MNB-FH + lex (Da Silva et al., 2014)	—	0.7508	0.7139	—	—	—	—
ENS(LR + RF + MNB)-FH (Da Silva et al., 2014)	—	0.5784	0.6517	—	—	—	—
ENS(LR + RF + MNB)-FH + lex (Da Silva et al., 2014)	—	0.7663	0.7456	—	—	—	—
WS-TSWE' [76]	0.841	—	—	—	—	—	—
WS-TSWE [76]	0.824	—	—	—	—	—	—
TSWE-P [76]	0.726	—	—	—	—	—	—
TSWE + P [76]	0.782	—	—	—	—	—	—
JSTH [76]	0.681	—	—	—	—	—	—
HTSM [76]	0.796	—	—	—	—	—	—
SAE (Pagliardini et al., 2018)	0.861	—	—	—	—	—	—
ParagraphVec DBOW (Pagliardini et al., 2018)	0.763	—	—	—	—	—	—
ParagraphVec DM (Pagliardini et al., 2018)	0.764	—	—	—	—	—	—
IST (Pu et al., 2019)	0.827	—	—	—	—	—	—
UST (Pu et al., 2019)	0.832	—	—	—	—	—	—
UIST (Pu et al., 2019)	0.845	—	—	—	—	—	—
RND	0.4895	0.4852	0.4847	0.5346	0.491	0.498	0.506
AFINN	0.601	0.674	0.3395	0.734	0.731	0.698	0.689
AFINN + RND	0.5455	0.5424	0.4475	0.6038	0.574	0.575	0.559
JST [8]	0.613	0.6176	0.4934	0.6398	0.611	0.605	0.579
TJST [8]	0.62	0.5898	0.4639	0.6814	0.645	0.616	0.614
RJST [12]	0.51	0.6086	0.4967	0.6232	0.609	0.546	0.561
TS [9]	0.5	0.5612	0.5251	0.565	0.54	0.545	0.547
WJST-1	0.821	0.7268	.75786	0.7126	0.779	0.76	0.737
WJST-2	0.818	0.7203	0.75775	0.7181	0.829	0.761	0.726
WJST1-1	0.8445	0.832	0.8	0.8317	0.796	0.77	0.773
WJST1-2	0.843	0.8352	0.7859	0.8373	0.798	0.774	0.769

TABLE 24: Sentiment classification in comparison with discriminative models on different datasets.

Method/Dataset	Android	Automotive	Electronic	Movie	
RND	0.48	0.4925	0.465	0.525	
AFINN	0.6975	0.625	0.675	0.595	
RND + AFINN	0.58	0.535	0.52	0.555	
LOGISTIC REGRESSION(BOW)	0.53	0.52	0.5338	0.57	
RANDOMFOREST(BOW)	0.69	0.65	0.60	0.6575	
SVM(BOW)	0.49	0.51	0.54	0.5675	
DECISIONTREE(BOW)	0.71	0.70	0.56	0.665	
NAIVE_BAYES(BOW)	0.51	0.55	0.4712	0.555	
KNEIGHBORS ( $N=3$ , BOW)	0.55	0.57	0.55	0.5725	
KNEIGHBORS ( $N=4$ ,BOW)	0.56	0.575	0.5915	0.585	
KNEIGHBORS ( $N=5$ ,BOW)	0.57	0.5725	0.55	0.59	
KNEIGHBORS ( $N=6$ ,BOW)	0.57	0.5675	0.56	0.61	
LOGISTIC REGRESSION (TF-IDF)	0.61	0.63	0.83	0.59	
RANDOMFOREST (TF-IDF)	0.575	0.6	0.7243	0.5225	
SVM (TF-IDF)	0.5825	0.59	0.80	0.54	
DECISIONTREE (TF-IDF)	0.55	0.57	0.7043	0.55	
NAIVE_BAYES (TF-IDF)	0.58	0.60	0.7945	0.58	
KNEIGHBORS ( $N=3$ , TF-IDF)	0.65	0.63	0.53	0.7125	
KNEIGHBORS ( $N=4$ , TF-IDF)	0.65	0.63	0.56	0.7125	
KNEIGHBORS ( $N=5$ , TF-IDF)	0.65	0.63	0.51	0.7125	
KNEIGHBORS ( $N=6$ , TF-IDF)	0.65	0.5725	0.53	0.7125	
JST	0.625	0.6575	0.7025	0.7575	
ASUM	0.613	0.6322	0.71	0.772	
TJST	0.765	0.7675	0.76	0.9475	
RJST	0.5825	0.615	0.5525	0.62	
TS	0.5425	0.5525	0.5475	0.5425	
WJST	Accuracy1	0.795	0.755	0.8625	0.8475
	Accuracy2	0.7825	0.7475	<b>0.875</b>	0.8325
WJST1	Accuracy1	<b>0.865</b>	<b>0.8</b>	0.8475	<b>0.97</b>
	Accuracy2	0.8525	0.795	0.855	0.9675

lexicon’s subjectivity. The trigram’s polarity will be opposed to the second or third word’s polarity if the first or second word is ‘not.’ The proposed methods are compared with JST on four datasets (single-domain) according to extended features (bigrams and trigrams) in the following experiment.

As shown in Table 25, the accuracy value is calculated on four datasets, and the experiment extends the features to bigrams and trigrams. According to the results, WJST1 outperforms WJST. According to evaluations results, proposed models outperform JST because the additional parameters can influence the process of producing words in a review appropriately. The perplexity value varies on different datasets because the size of datasets is different. As the number of grams increases, perplexity is increased because in each document, in addition to the unigrams (+bigrams), bigrams (+trigrams) are added to the data, and the size of the dataset is increased. As the number of grams increases, accuracy is improved. In some cases, it gets worse because higher grams (bigram or trigrams) are sometimes meaningless.

*4.13. Discussions on the Limitations of the Proposed Methods.* Although the analysis of the results of the evaluation can demonstrate the best performance of proposed methods, proposed methods have some limitations, as follows:

- (1) The first limitation is the time complexity of the proposed methods ( $O(G \cdot w_{ALL} \cdot |S| \cdot |Z| \cdot |Q| \cdot |E|)$ ) which is more than baseline methods ( $O(G \cdot w_{ALL} \cdot |S| \cdot |Z|)$ ) according to Table 3 in Section 3.3.
- (2) The second limitation is the window size. On the report in Table 13, an increase in the size of a window will decrease the accuracy because it will increase the number of words in the window, and each term may not affect all neighbors in its window. Therefore, finding the words that can be involved in a window is a new challenge that we introduce in this study, and two methods are presented in Section 4.7. The first method assumes that each word affects all neighbors in its window, and the second method assumes that each word affects some random neighbors in its window. Therefore, the first method selects all neighbors, but the second method selects some neighbors randomly. According to the results, the second method is more stable than the first method in terms of accuracy, but the first method has a higher accuracy value than the second method. The second method outperforms the first method in terms of perplexity. The first method performs better than the second in terms of accuracy and topic\_coherency.

TABLE 25: Sentiment classification according to extended features (bigrams and trigrams) on different datasets (single-domain).

Model	Gram	Metric\dataset	Android	Automotive	Electronic	Movie
JST	U *	Accuracy	0.625	0.6275	0.675	0.7575
		Perplexity	19.6726	24.7154	25.227	27.3111
		Topic_Coh	-4.6026	-1.486	-1.5892	-1.0123
	U + B*	Accuracy	0.655	0.7025	0.75	0.835
		Perplexity	66.1882	86.6897	81.3375	98.9201
		Topic_Coh	-2.4595	-1.0417	-0.9823	-0.4165
	U + B + T*	Accuracy	0.6775	0.5825	0.7525	0.7275
		Perplexity	114.8812	158.7113	156.1383	190.8403
		Topic_Coh	-1.5776	-2.3915	-0.7030	-0.1288
	WJST	U	Accuracy1	0.7925	0.755	0.8625
Accuracy2			0.7775	0.7475	0.875	0.71
Perplexity			16.7303	21.2008	20.5489	23.1145
Topic_Coh			-0.5547	-1.6542	-2.0442	-0.0751
U + B		Accuracy1	0.6825	0.63	0.67	0.65
		Accuracy2	0.68	0.625	0.6675	0.6475
		Perplexity	36.5505	49.8584	48.2008	58.0838
		Topic_Coh	-1.3976	-4.0214	-1.2513	-0.0423
U + B + T		Accuracy1	0.7325	0.6825	0.635	0.6475
		Accuracy2	0.7325	0.6825	0.615	0.6325
		Perplexity	52.9205	72.7629	76.2545	88.1860
		Topic_Coh	-1.3242	-1.5525	-1.2409	-2.5170
WJST1	U	Accuracy1	0.81	0.80	0.79	0.97
		Accuracy2	0.7925	0.79	0.80	0.9625
		Perplexity	16.6787	21.4357	22.6012	24.6302
		Topic_Coh	-0.187	-1.1883	-1.1683	-0.1348
	U + B	Accuracy1	0.7	0.7625	0.8575	0.935
		Accuracy2	0.7	0.7625	0.8525	0.935
		Perplexity	38.5603	49.5167	54.7041	62.3459
		Topic_Coh	-2.0962	-1.8607	-0.9744	-0.0617
	U + B + T	Accuracy1	0.83	0.7625	0.745	0.9475
		Accuracy2	0.8275	0.7675	0.745	0.9475
		Perplexity	55.5819	75.3624	77.6660	96.3675
		Topic_Coh	-1.3837	-0.1840	-0.9334	-0.0333

U\*: unigram. B\*: bigram. T\*: trigram.

**4.14. A Concise Description of the Proposed Solutions and the Results.** The main problem in this study is to examine a user's opinion about a product or movie, for example. This means identifying whether a user has a positive or negative idea about a subject (a product or movie).

Two novel models have been proposed that use topic modeling to solve the above problem. So, our solution is using a technique named topic modeling. Proposed models extend and improve JST (as a topic model) through two new parameters. To improve JST, proposed models consider the effect of words on each other. The new parameters have an immense effect on model accuracy regarding evaluation results. According to evaluations results, the proposed models outperform JST because the additional parameters can influence the process of producing words in a review appropriately. They can improve sentiment classification at the document-level. Also, in the evaluation results report, proposed methods are more accurate than discriminative models such as SVM and logistic regression. Proposed methods are more flexible than discriminative models because other information, such as the top 10 words, can be extracted from the heart of the data.

## 5. Conclusion

In this study, two new models called WJST and WJST1 have been presented that extend JST and improve accuracy metrics. Reviewing the various articles about sentiment analysis indicates that the proposed models are associated with innovation and lead to remarkable results compared to the baseline methods. The proposed models can generate a sentiment dictionary. According to evaluations results, the proposed models consider the effect of words on each other using the extra parameters, which are important and influential. The evaluation results indicate that the accuracy has been improved compared to the baseline methods such as JST, RJST, TS, TJST, and ALGA. Results show that the proposed methods perform better with different topic number settings. WJST1 outperforms other methods in terms of accuracy, demonstrating its effectiveness of that. Prior sentiment information affects perplexity and topic\_coherency lower than accuracy.

According to evaluations results, using the AFINN dictionary as prior sentiment information is more effective than using the NO\_AFINN state. ALGA uses the genetic algorithm to generate a sentiment dictionary; however,

proposed methods use topic modeling to generate this dictionary. According to the evaluation results, the proposed models outperform JST because the additional parameters could influence the process of producing words in a review appropriately. They have the potential to increase the emotion detection's accuracy at the level of the document. The proposed methods are unsupervised, and no labeled data is required. Proposed methods can automatically assess web comments and categorize reviews as positive or negative. The proposed methods have tried to increase the accuracy with fewer parameters and, at the same time, simplicity compared to the existing methods. The proposed methods both analyze emotions at the document-level and create an emotional dictionary. They are also the first methods to create an emotional dictionary through a topic modeling technique and in an automatic and accurate way. The proposed methods are the first methods that consider the words in the text and their effect on each other in a dynamic and weighty way. Also, they are parametric.

## 6. Future Work

The proposed models are parametric in the present study, and further studies will be conducted to investigate non-parametric models. Sentiment classification on multidomain datasets is a challenge, and further studies can be conducted to investigate this problem for future research. In future research, the proposed methods can be evaluated on more datasets. More parameters can also be assessed. Twitter social network data have obtained significant attention in natural language processing studies, with certain conditions, such as short data length. In future articles, the proposed methods can be modified to analyze emotions with specific Twitter data.

## Data Availability

The datasets used during the current study are available from the corresponding author (a.osmani@qiau.ac.ir) upon reasonable request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] N. Peikari, A. Yaghobi, and R. Taheri, *Analysis of Emotions on the Twitter Social Network with the Text-Mining Technique*, International Conference on Web Research, Iran, 2015.
- [2] L. Zhang and B. Liu, "Sentiment analysis and opinion mining," in *Encyclopedia of Machine Learning and Data Mining*, C. Sammut and G. I. Webb, Eds., Springer, Illinois Chicago, 2017.
- [3] B. Pang and L. Lee, "A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts," in *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pp. 271–278, ACL, Barcelona Spain, July 2004.
- [4] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Foundations and Trends in Information Retrieval*, vol. 2, no. 1–2, pp. 1–135, 2008.
- [5] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: a survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093–1113, 2014.
- [6] Z. Chen and B. Liu, *Lifelong Machine Learning*, Morgan & Claypool Publishers, San Rafael, California USA, 2018.
- [7] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *Proceedings of the National Academy of Sciences*, vol. 101, no. suppl\_1, pp. 5228–5235, 2004.
- [8] C. Lin and Y. He, "Joint sentiment/topic model for sentiment analysis," in *Proceedings of the 18th ACM conference on Information and knowledge management*, ACM, pp. 375–384, New York, NY, USA, November 2009.
- [9] M. Dermouche, L. Kouas, J. Velcin, and S. Loudcher, "A joint model for topic-sentiment modeling from text," in *Proceedings of the 30th Annual ACM Symposium On Applied Computing*, ACM, pp. 819–824, New York, NY, USA, April 2015.
- [10] C. Lin, Y. He, R. Everson, and S. Ruger, "Weakly supervised joint sentiment-topic detection from text," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, pp. 1134–1145, 2012.
- [11] B. Liu, "Sentiment analysis and opinion mining," *Synthesis Lectures on Human Language Technologies*, vol. 5, no. 1, pp. 1–167, 2012.
- [12] B. Liu and L. Zhang, "A survey of opinion mining and sentiment analysis," in *Mining Text Data*, C. Aggarwal and C. Zhai, Eds., Springer, Boston, MA, 2012.
- [13] Z. Yuan, S. Wu, F. Wu, J. Liu, and Y. Huang, "Domain attention model for multi-domain sentiment classification," *Knowledge-Based Systems*, vol. 155, pp. 1–10, 2018.
- [14] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.
- [15] X. Li, A. Zhang, C. Li, L. Guo, W. Wang, and J. Ouyang, "Relational biterm topic model: short-text topic modeling using word embeddings," *The Computer Journal*, vol. 62, no. 3, pp. 359–372, 2018.
- [16] M. Yang, X. Chen, W. Tu, Z. Lu, J. Zhu, and Q. Qu, "A topic drift model for authorship attribution," *Neurocomputing*, vol. 273, pp. 133–140, 2018.
- [17] J.-T. Chien, C.-H. Lee, and Z.-H. Tan, "Latent Dirichlet mixture model," *Neurocomputing*, vol. 278, pp. 12–22, 2018.
- [18] A. Osmani, JB. Mohasefi, and FS. Gharehchopogh, "Enriched latent dirichlet allocation for sentiment analysis," *Expert Systems*, vol. 37, Article ID e12527, 2020.
- [19] C. Lin, E. Ibeke, A. Wyner, and F. Guerin, "Sentiment-topic modeling in text mining," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 5, no. 5, pp. 246–254, 2015.
- [20] Y. Jo and A. H. Oh, "Aspect and sentiment unification model for online review analysis," ACM, in *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, pp. 815–824, Hong Kong, China, February 2011.
- [21] J. A. Wahid, L. Shi, Y. Gao et al., "Topic2features: a novel framework to classify noisy and sparse textual data using LDA topic distributions," *PeerJ Computer Science*, vol. 7, p. e677, 2021.
- [22] Y. Zhou, L. Liao, Y. Gao, R. Wang, and H. Huang, "TopicBERT: a topic-enhanced neural language model fine-tuned for sentiment classification," in *Proceedings of the IEEE Transactions on Neural Networks and Learning Systems*, New York, NY, USA, August 2021.

- [23] S. Abdi, J. Bagherzadeh, G. Gholami, and M. S. Tajbakhsh, "Using an auxiliary dataset to improve emotion estimation in users' opinions," *Journal of Intelligent Information Systems*, vol. 56, no. 3, pp. 581–603, 2021.
- [24] M. H. Alam, W.-J. Ryu, and S. Lee, "Joint multi-grain topic sentiment: modeling semantic aspects for online reviews," *Information Sciences*, vol. 339, pp. 206–223, 2016.
- [25] T. Hua, C.-T. Lu, J. Choo, and C. K. Reddy, "Probabilistic topic modeling for comparative analysis of document collections," *ACM Transactions on Knowledge Discovery from Data*, vol. 14, no. 2, pp. 1–27, 2020.
- [26] Y. Rao, Q. Li, X. Mao, and L. Wenyan, "Sentiment topic models for social emotion mining," *Information Sciences*, vol. 266, pp. 90–100, 2014.
- [27] A. Mahadevan and M. Arock, "Integrated topic modeling and sentiment analysis: a review rating prediction approach for recommender systems," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 28, no. 1, pp. 107–123, 2020.
- [28] Y. Xia, G. Tang, H. Zhao, E. Cambria, and T.-F. Zheng, "Using word sense as a latent variable in LDA can improve topic modeling," in *Proceedings of the 6th International Conference on Agents and Artificial Intelligence*, pp. 532–537, Science and Technology Publications, Loire Valley France, January 2014.
- [29] Y. Meng, W. F. Speier, M. Ong, and C. Arnold, "HCET: hierarchical clinical embedding with topic modeling on electronic Health Record for predicting depression," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 4, pp. 1–8, 2020.
- [30] J. Qiang, Z. Qian, Y. Li, Y. Yuan, and X. Wu, "Short text topic modeling techniques, applications, and performance: a survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, pp. 1–19, 2020.
- [31] Q. Yang, Y. Rao, H. Xie, J. Wang, F. L. Wang, and W. H. Chan, "Segment-level joint topic-sentiment model for online review analysis," *IEEE Intelligent Systems*, vol. 34, no. 1, pp. 43–50, 2019.
- [32] I. Vayansky and S. A. P. Kumar, "A review of topic modeling methods," *Information Systems*, vol. 94, pp. 1–32, 2020.
- [33] E. Bigne, C. Ruiz, A. Cuenca, C. Perez, and A. Garcia, "What drives the helpfulness of online reviews? A deep learning study of sentiment analysis, pictorial content and reviewer expertise for mature destinations," *Journal of Destination Marketing & Management*, vol. 20, 2021.
- [34] Q. Safder, Z. Mahmood, R. Sarwar et al., "Sentiment analysis for Urdu online reviews using deep learning models," *Expert Systems*, 2021.
- [35] AR. Pathak, M. Pandey, and S. Rautaray, "Topic-level sentiment analysis of social media data using deep learning," *Applied Soft Computing*, vol. 108, 2021.
- [36] S. Feng, H. Zhou, and H. Dong, "Using deep neural network with small dataset to predict material defects," *Materials & Design*, vol. 162, pp. 300–310, 2019.
- [37] S. Khezri, K. Faez, and A. Osmani, "Modified discrete binary PSO based sensor placement in WSN networks," in *Proceedings of the 2010 International Conference on Computational Intelligence and Communication Networks*, pp. 200–204, Bhopal, India, November 2010.
- [38] S. Khezri, M. R. Meybodi, and A. Osmani, "Fuzzy adaptive PBIL based sensor placement in wireless sensor networks," in *Proceedings of the 2011 International Symposium on Computer Networks and Distributed Systems*, pp. 216–221, CNDS), Tehran, Iran, February 2011a.
- [39] S. Khezri, K. Faez, and A. Osmani, "An intelligent sensor placement method to reach a high coverage in wireless sensor networks," *International Journal of Grid and High Performance Computing*, vol. 3, no. 3, pp. 54–68, 2011b.
- [40] F. S. Gharehchopogh and H. Gholizadeh, "A comprehensive survey: whale Optimization Algorithm and its applications," *Swarm and Evolutionary Computation*, vol. 48, pp. 1–24, 2019.
- [41] B. Abdollahzadeh, F. S. Gharehchopogh, and S. Mirjalili, "African vultures optimization algorithm: a new nature-inspired metaheuristic algorithm for global optimization problems," *Computers & Industrial Engineering*, vol. 158, 2021a.
- [42] B. Abdollahzadeh, F. Soleimani Gharehchopogh, and S. Mirjalili, "Artificial gorilla troops optimizer: a new nature-inspired metaheuristic algorithm for global optimization problems," *International Journal of Intelligent Systems*, vol. 36, no. 10, pp. 5887–5958, 2021b.
- [43] K. Asghari, M. Masdari, F. S. Gharehchopogh, and R. Saneifard, "Multi-swarm and chaotic whale-particle swarm optimization algorithm with a selection method based on roulette wheel," *Expert Systems*, vol. 38, no. 8, 2021.
- [44] R. Asgarneshad, S. A. Monadjemi, and M. Soltanaghaei, "An application of MOGW optimization for feature selection in text classification," *The Journal of Supercomputing*, vol. 77, no. 6, pp. 5806–5839, 2021.
- [45] A. K. Abasi, A. T. Khader, M. A. Al-Betar, S. Naim, S. N. Makhadmeh, and Z. A. A. Alyasseri, "An improved text feature selection for clustering using binary grey wolf optimizer," in *Proceedings of the 11th National Technical Seminar on Unmanned System Technology 2019. Lecture Notes in Electrical Engineering*, Z. Zain, Ed., vol. 666, Singapore, Springer, 2021.
- [46] A. Osmani, J. B. Mohasefi, and F. S. Gharehchopogh, "Sentiment classification using two effective optimization methods derived from the artificial bee colony optimization and imperialist competitive algorithm," *The Computer Journal*, vol. 65, no. 1, pp. 18–66, 2022.
- [47] M. Tang, J. Jin, Y. Liu, C. Li, and W. Zhang, "Integrating topic. Sentiment. And syntax for modeling online reviews: a topic model approach," *ASME. J. Comput. Inf. Sci. Eng.*, vol. 19, no. 1, pp. 1–12, 2018.
- [48] L. Gong and H. Wang, "When sentiment analysis meets social network: a holistic user behavior modeling in opinionated data," in *Proceedings of the 24th acm sigkdd international conference on knowledge discovery & data mining (kdd '18)*, pp. 1455–1464, Association for Computing Machinery (ACM), London, United Kingdom, July 2018.
- [49] K. Cheng, J. Li, J. Tang, and H. Liu, "Unsupervised Sentiment Analysis with Signed Social Networks," in *Proceedings of the AAAI Conference on Artificial Intelligence, 31(1), Association for the Advancement of Artificial Intelligence (AAAI Press)*, pp. 3429–3435, San Francisco, California, USA, 2017.
- [50] M. Iqbal, A. Karim, and F. Kamiran, "Balancing prediction errors for robust sentiment classification," *ACM Transactions on Knowledge Discovery from Data*, vol. 13, no. 3, pp. 1–21, 2019.
- [51] M. Ibrahim Hussein, X. Kang, L. Feng, Z. Ibrahim, A. Waheed Ahmed, and Q. Guilin, "A cross-lingual sentiment topic model evolution over time. A Cross-lingual Sentiment Topic Model Evolution over Time," *Intelligent Data Analysis*, vol. 24, no. 2, pp. 253–266, 2020.
- [52] S. Kumar, R. Singh, M. Z. Khan, and A. Noorwali, "Design of adaptive ensemble classifier for online sentiment analysis and opinion mining," *PeerJ Computer Science*, vol. 7, 2021.

- [53] H. Keshavarz and M. S. Abadeh, "ALGA: adaptive lexicon learning using genetic algorithm for sentiment analysis of microblogs," *Knowledge-Based Systems*, vol. 122, pp. 1–16, 2017.
- [54] P. Kalarani and S. Selva Brunda, "Sentiment analysis by POS and joint sentiment topic features using SVM and ANN," *Soft Computing*, vol. 23, no. 16, pp. 7067–7079, 2019.
- [55] A. Farkhod, A. Abdusalomov, F. Makhmudov, and Y. I. Cho, "LDA-based topic modeling sentiment analysis using topic/document/sentence (TDS) model," *Applied Sciences*, vol. 11, no. 23, p. 11091, 2021.
- [56] M. Fatemi and M. Safayani, "Joint sentiment/topic modeling on text data using a boosted restricted Boltzmann Machine," *Multimedia Tools and Applications*, vol. 78, no. 15, pp. 20637–20653, 2019.
- [57] Q. Liang, S. Ranganathan, K. Wang, and X. Deng, "JST-RR model: joint modeling of ratings and reviews in sentiment-topic prediction," 2021, <https://arxiv.org/abs/2102.11048>.
- [58] N. Pathik and P. Shukla, "An efficient sentiment analysis using topic model based optimized recurrent neural network," *International Journal on Smart Sensing and Intelligent Systems*, vol. 14, no. 1, pp. 1–12, 2021.
- [59] A. Sengupta, W. S. Paka, S. Roy, G. Ranjan, and T. Chakraborty, "An embedding-based joint sentiment-topic model for short texts," *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 15pp. 633–6431, Seattle, USA, 30 March 2021.
- [60] L. Chen and S. Mankad, "A structural topic sentiment model for text analysis," vol. 2022, 2022, <https://ssrn.com/abstract>.
- [61] F. Huang, X. Li, C. Yuan, S. Zhang, J. Zhang, and S. Qiao, "Attention-emotion-enhanced convolutional LSTM for sentiment analysis," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.
- [62] B. Özyurt and M. A. Akcayol, "A new topic modeling based approach for aspect extraction in aspect based sentiment analysis: SS-LDA," *Expert Systems with Applications*, vol. 168, Article ID 114231, 2021.
- [63] W. Zhao, Z. Guan, L. Chen et al., "Weakly-supervised deep embedding for product review sentiment analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 1, pp. 185–197, 2017.
- [64] G. Rao, W. Huang, Z. Feng, and Q. Cong, "LSTM with sentence representations for document-level sentiment classification," *Neurocomputing*, vol. 308, pp. 49–57, 2018.
- [65] U. Naseem, I. Razzak, K. Musial, and M. Imran, "Transformer based deep intelligent contextual embedding for twitter sentiment analysis," *Future Generation Computer Systems*, vol. 113, pp. 58–69, 2020.
- [66] M. E. Basiri, S. Nemati, M. Abdar, E. Cambria, and U. R. Acharya, "ABCDM: an attention-based bidirectional CNN-RNN deep model for sentiment analysis," *Future Generation Computer Systems*, vol. 115, pp. 279–294, 2021.
- [67] F. Tang, L. Fu, B. Yao, and W. Xu, "Aspect based fine-grained sentiment analysis for online reviews," *Information Sciences*, vol. 488, no. 2019, pp. 190–204, 2019.
- [68] Z. Yang, A. Kotov, A. Mohan, and S. Lu, "Parametric and non-parametric user-aware sentiment topic models," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 413–422, ACM, Santiago, Chile, August 2015.
- [69] F. Å Nielsen, "A new ANEW: evaluation of a word list for sentiment analysis in microblogs," in *Proceedings of the ESWC2011 Workshop on "Making Sense of Microposts": Big Things Come in Small Packages*, pp. 93–98, ESWC2, Heraklion, Crete, May 2011.
- [70] J. Blitzer, M. Dredze, and F. Pereira, "Biographies, bollywood, boom-boxes and blenders: domain adaptation for sentiment classification," in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp. 440–447, prague czech republic, June 2007.
- [71] D. Kotzias, M. Denil, N. . d. Freitas, and P. Smyth, "From group to individual labels using deep features," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 597–606, Sydney, NSW, Australia, August 2015.
- [72] N. Proelochs, S. Feuerriegel, and D. Neumann, "Statistical inferences for polarity identification in natural language," *PLoS One*, vol. 13, no. 12, Article ID e0209323, 2017.
- [73] M. Hu and B. Liu, "Mining and summarizing customer reviews," . in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 168–177, ACM, Seattle, WA, USA, August 2004.
- [74] B. Liu, M. Hu, and J. Cheng, "Opinion observer: analyzing and comparing opinions on the web," in *Proceedings of the 14th International Conference on World Wide Web*, pp. 342–351, ACM, Chiba, Japan, May 2005.
- [75] S. Moghaddam and M. Ester, "ILDA: interdependent LDA model for learning latent aspects and their ratings from online product reviews," in *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 665–674, Beijing, China, January 2011.
- [76] Y. Bao and A. Datta, "Simultaneously discovering and quantifying risk types from textual risk disclosures," *Management Science*, vol. 60, no. 6, pp. 1371–1391, 2014.
- [77] M. Shams and A. Baraani-Dastjerdi, "Enriched LDA (ELDA): combination of latent Dirichlet allocation with word co-occurrence analysis for aspect extraction," *Expert Systems with Applications*, vol. 80, pp. 136–146, 2017.
- [78] D. Mimno, H.-M. Wallach, E. Talley, M. Leenders, and A. McCallum, "Optimizing semantic coherence in topic models," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 262–272, EMNLP, Edinburgh, United Kingdom, July 2011.
- [79] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *Journal of the American Statistical Association*, vol. 32, pp. 674–701, 1937.
- [80] M. Friedman, "A comparison of alternative tests of significance for the problem of  $m$  rankings," *The Annals of Mathematical Statistics*, vol. 11, no. 1, pp. 86–92, 1940.
- [81] X. Fu, X. Sun, H. Wu, L. Cui, and J. Z. Huang, "Weakly supervised topic sentiment joint model with word embeddings," *Knowledge-Based Systems*, vol. 147, pp. 43–54, 2018.
- [82] M. Pagliardini, P. Gupta, and M. Jaggi, "Unsupervised Learning of Sentence Embeddings Using Compositional n-Gram Features," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 528–540, New Orleans, Louisiana, vol. 1, (Long Papers), 2018.
- [83] X. Pu, G. Wu, and C. Yuan, "User-aware topic modeling of online reviews," *Multimedia Systems*, vol. 25, pp. 59–69, 2019.
- [84] N-F-F. Da Silva, E-R. Hruschka, and E-R. Hruschka, "Tweet sentiment analysis with classifier ensembles," *Decision Support Systems*, vol. 66, pp. 170–179, 2014.
- [85] H. Saif, Y. He, M. Fernandez, and H. Alani, "Semantic Patterns for Sentiment Analysis of Twitter," in *The Semantic Web*

- ISWC 2014. *ISWC 2014 Lecture Notes in Computer Science*, et al. Vol. vol 8797, Springer, Cham, Italy2014.
- [86] M. Mirhosseini and H. BICA Nezamabadi-pour, “a binary imperialist competitive algorithm and its application in CBIR systems,” *Int. J. Mach. Learn. Cybern*, vol. 28, no. 1–15, 2017.
- [87] L. Shang, Z. Zhou, and X. Liu, “Particle swarm optimization-based feature selection in sentiment classification,” *Soft Comput*, vol. 20, pp. 3821–3834, 2016.
- [88] M. Schiezero and H Pedrini, “Data feature selection based on artificial bee colony algorithm,” *EURASIP J. Image Video Process*, vol. 47, no. 1–8, 2013.



## Research Article

# Entropy and Gaussian Filter-Based Adaptive Active Contour for Segmentation of Skin Lesions

**Saleem Mustafa,<sup>1</sup> Muhammad Waseem Iqbal<sup>2</sup>, Toqir A. Rana<sup>3,4</sup>, Arfan Jaffar,<sup>1</sup> Muhammad Shiraz,<sup>5</sup> Muhammad Arif,<sup>3</sup> and Samia Allaoua Chelloug<sup>6</sup>**

<sup>1</sup>Department of Computer Science, Superior University, Lahore 54600, Pakistan

<sup>2</sup>Department of Software Engineering, Superior University, Lahore 54600, Pakistan

<sup>3</sup>Department of Computer Science and IT, The University of Lahore, Lahore 54000, Pakistan

<sup>4</sup>School of Computer Sciences, Universiti Sains Malaysia, Penang 11800, Malaysia

<sup>5</sup>Department of Computer Science, Federal Urdu University of Arts, Science & Technology, Islamabad 44000, Pakistan

<sup>6</sup>Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P. O. Box 84428, Riyadh 11671, Saudi Arabia

Correspondence should be addressed to Samia Allaoua Chelloug; [sachelloug@pnu.edu.sa](mailto:sachelloug@pnu.edu.sa)

Received 14 April 2022; Revised 13 June 2022; Accepted 28 June 2022; Published 19 July 2022

Academic Editor: Abdul Rehman Javed

Copyright © 2022 Saleem Mustafa et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Malignant melanoma is considered one of the deadliest skin diseases if ignored without treatment. The mortality rate caused by melanoma is more than two times that of other skin malignancy diseases. These facts encourage computer scientists to find automated methods to discover skin cancers. Nowadays, the analysis of skin images is widely used by assistant physicians to discover the first stage of the disease automatically. One of the challenges the computer science researchers faced when developing such a system is the un-clarity of the existing images, such as noise like shadows, low contrast, hairs, and specular reflections, which complicates detecting the skin lesions in that images. This paper proposes the solution to the problem mentioned earlier using the active contour method. Still, seed selection in the dynamic contour method has the main drawback of where it should start the segmentation process. This paper uses Gaussian filter-based maximum entropy and morphological processing methods to find automatic seed points for active contour. By incorporating this, it can segment the lesion from dermoscopic images automatically. Our proposed methodology tested quantitative and qualitative measures on standard dataset dermis and used to test the proposed method's reliability which shows encouraging results.

## 1. Introduction

Melanoma is one form of skin cancer. Recent researches show that melanoma is the most dangerous kind of skin cancer. An important reason is that melanoma affects about 75% of death reported skin cancer. In 2017, research shows that around 9,480 out of 76,690 melanoma patients died by the cause of melanoma in the USA [1]. In addition, 1 in 74 males and 1 in 90 females may infect with melanoma in their life in Canada. Previous research also shows that for non-Hispanic American white people, occurrence rates have

increased yearly by around 3%. For grown people ages 15 and 30, melanoma is considered the most popular identifiable type of cancer [2]. Discovering melanoma in the early stage will increase the probability from 5 years to 96% of remaining alive. Still, in case of finding it in the very advanced stage, that percentage will decrease to 5% [2]. The recovery percentage is affected, and the melanoma treatment cost in the advanced stage is 30 times more than the cost of melanoma treatment in the early stages. Dermoscopic device helps physicians view the lesion features more clearly than the naked eye [3].

Diagnosis of skin lesion and classification has been an activity in most areas of medical science research for quite a while now. Melanoma is one of the lethal and rare forms of skin lesion. Due to this type of lesion, the death rate is three times more than all other skin lesions. It is common, especially among white skin people. Diagnosis of malignant melanoma in the initial stage increases the survival rates and is curable. Unluckily, screening all the patients with a skin lesion for dermatologists is very difficult, and the cost of treating this type of skin lesion is very high. Working out the dermoscopic clinical features like lesion borders, pigment networks, and the color of melanoma is a vital step for dermatologists, who require a method for the correct clinical diagnosis and ensure the proper treatment. These structures are considered one of the primary keys that detect and classify melanoma or nonmelanoma disease. Hence, we need an intelligent automated system to diagnose a patient's danger of malignant melanoma using dermoscopic images.

The need for a computerized skin cancer detection method is becoming one of the most challenging sciences. It is because of the detecting process, which is quite tricky. Also, the existing skin cancer images may contain noise such as hairs, shadows, specular reflections, and low contrast that reduce the image quality, which causes the accuracy of skin lesion segmentation algorithms to be less than clear images [4]. Active contour is a well-known method that can be used for segmentation. The active contour's primary purpose is to evolve a segmentation curve [5]. For example, to segment an object within the image, the curve of active contour starts from the initial point until it meets the object's boundary. Hence, starting with the angle around the object is the foremost important point in this method. It is the main drawback of the active contour method. This initialization problem also leads to low performance for concave boundaries and low performance when the contour is started far from the minimum. Hence, one of the main disadvantages of localizing active contour is the sensitive initial boundary to segment the object in the image. Some existing techniques proposed by hand allocate the initial point of seed to start active contour. Several methods carry out restarting if the first start does not return the object's accurate boundary of the object, so this is computationally expensive. The re-initialization process will maintain single object segmentation, while the initialization method is multiple object segmentation. However, the initialization method provides faster computational time than re-initialization. Therefore, we have proposed a method for initializing the seed point automatically, and it becomes more efficient and faster computational time than re-initialized. Sometimes, it also returns more than one object inside each other, which is another drawback.

This research paper will review existing techniques and propose a new skin cancer segmentation method.

**1.1. Skin Anatomy Overview.** The skin and its certain specialized derivatives called appendages constitute the integumentary system. The appendages include sweat glands, hairs, sebaceous glands, and nails. The skin (also called

integument or cutis) forms the external covering of the body. It consists of two layers of completely different types of tissues attached over their entire extent. The superficial layer known as the *epidermis* consists of stratified squamous epithelium, while the deeper layer, called the *dermis*, is composed of dense connective tissue. Together, these two layers form a sheet that varies from 0.5 mm to 4.0 mm in thickness in different body parts [6]. The skin is generally classified into thick and thin types. The thick type covers palms and soles, while the thin skin is found on the remainder of the body. It is important to note that these terms, thick and thin, take into account the thickness of the *epidermis* only and do not refer to the thickness of the skin as a whole. Figure 1 shows the basics of skin layers.

**1.2. Epidermis.** The *epidermis* is a continuously self-replacing laminated squamous keratinized epithelium. It contains four types of cells: melanocyte cells, keratinocyte cells, Langerhans cells, and Merkel cells. Figure 2 shows the *epidermis* layers and structures.

**1.3. Melanoma Growth Phase.** The path of melanoma usually pursues two growth phases [7], namely, the vertical growth phase (VGP) and the radial growth phase (RGP). RGP generally leads to vertical growth phase. During the radial growth phase, cancerous cells spread outward in a radial pattern through the *epidermis* [8, 9]. At this moment, melanin and melanocytes are still constrained to the *epidermis* (that is to say, the skin cancer has not metastasized), so it is considered "in situ." In order to achieve a favorable prognosis, this malignant cancer should be detected earlier when it is in its radial growth phase, earlier than cancer goes into a metastatic vertical growth phase. In VGP, melanocytes enter the *dermis* deeply and may invade the surrounding body tissues through metastatic events. This invasion of distant tissues is precarious since cancer can go wide in the body in several different components of the body [10, 11]. Lesions within the vertical growth phase frequently emerge as a nodule or bump on the surface. It is a hazardous phase, so lesions within this phase need to be immediately addressed.

## 2. Related Work

Regardless of current research to design decision support systems, there is much room to facilitate the scientists to explore and suggest efficient and effective decision support systems for acknowledgment in the field of medicine [12–14]. The method for the detection in the early stages is mainly concerned with the clinical characteristics utilized for image processing. These systematic characteristics provide valuable information to medical specialists, which assists them in analyzing melanoma patients. Computer-aided design (CAD) systems help physicians and practitioners to perform their responsibilities efficiently [15–17]. CAD systems are designed using visual features of images, but in selecting suitable features, major problems arise during the feature extraction stage. Therefore, this problem will be

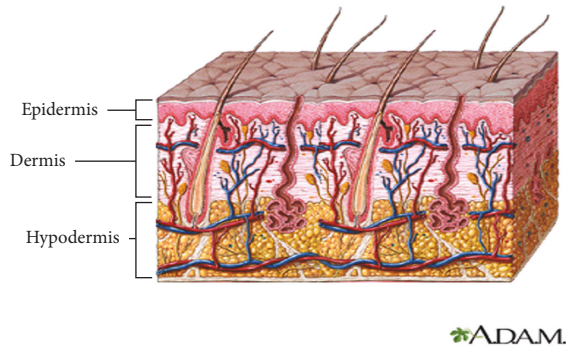


FIGURE 1: Skin layers.

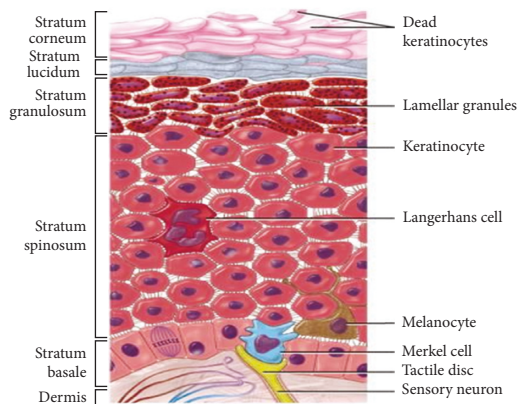


FIGURE 2: Epidermis layers and structures.

solved in our proposed research work to segment and classify the visual characteristics effectively.

Many techniques have been presented to segment skin lesion images automatically, but existing algorithms are only proposed for illumination. Segmentation procedures can be helpful to only those dermoscopy images, where there is a lot of difference between specific kinds of cancer and the surrounding skin area. Nawaz et al. [18] gathered a summary of new standard segmentation algorithms for dermoscopy images. The algorithms are compared with simple thresholding and region merging using the active contour method [19]. Many systems use only those characteristics obtained from pixel color [20]. The luminance channel from the CIE 1976  $L^*a^*b^*$  (CIELAB) or CIE 1976  $L^*u^*v^*$  (CIELUV) color spaces, the blue channel from the RGB color space, and an orthogonal transform are applied to these color channels. But, it is still a problem to segment the lesion parts correctly with undistinguishable boundaries when we rely only on color features. Such a method is not being operated on images of skin cancer.

In addition, some segmentation texture-based algorithms have been applied to dermoscopic images. Öztürk and Özkaya [21] have used some basic numerical methods, such as the gray-level co-occurrence matrix (GLCM), to recognize the texture or pattern of images of the skin body. This analysis can be helpful for the segmentation of the lesion parts and the classification of dermoscopic images. A textural-based algorithm for segmentation lesions using

GLCM features was presented in [22]. Similarly, first-order region statistics and Sonali random field models were introduced. Sonali and Kamat [23] proposed a new method by using the natural skin texture sample on the four edges of the dermoscopic image. The presence of artifacts like shining areas and shadows caused by light makes the segmentation of skin lesion images more complicated. Hence, Manjubharathi and Saraswathi [24] explored more different algorithms, including thresholding, split-and-merge, and active contours, to change them to be specific for lesion images. The thresholding algorithm has to be changed to account for shining areas. Jeniva and Santhi [25] provided four different algorithms that modify the lighting variant of images before thresholding is applied to segmentation methods. Thresholding is used on multiple colors, a group of similar colors, or one-color channels derived by processing steps and primary component analysis [26]. Abbadi and Miry [27] proposed segmentation techniques, including the preprocessing stage, which assist in performing segments. Lu et al. [28] joined thresholding approach of segmentation for boundary lesion image with the form of fuzzy C-means to last segmentation process. Another algorithm was proposed based on the analysis of the NC ratio in segmentation by Baral, Gonnade, and Verma [29].

The experimental result shows high efficiency and robust segmentation process for a tumor cell. Khoulood et al. [30] applied the idea of texture distribution based on the learned model of natural human skin and cancer textures. Their results represented the difference between the texture distribution metric and captured the pair of texture distribution. The image is divided into many smaller parts based on similarity to achieve higher segmentation. Marchetti et al. [31] proposed the segmentation method to identify skin lesion cells in the superficial. The nuclei regions located on the superficial layer of skin are segmented with a  $k$  value equal to 3 using the K-means clustering techniques based on some color and space information [32]. Then, local region recursive technique segmentation is used to filter the interested nuclei regions to identify the area of nuclei which uses the intensity and size of nuclei as a parameter [33]. Finally, in the last stage, the local double ellipse descriptor is applied to discriminate melanocyte cells from keratinocyte cells.

The performance of the technique produced good results where the foreground and background both have the same form. Zhang et al. [34] suggested another Wiener filter technique to remove artifacts such as skin hair from the image. Then, they applied to thresholding to segment the skin lesion parts from the dermoscopic image. Experts in the medical domain compare the results of this segmentation method and measure the distance between these two results by using 96.32 percent accuracy of TDR and HM [33]. Gonzalez-Diaz [35] suggested segmenting melanocytes from the histopathological picture of the body. First, the local region's recursive algorithm and mean shift are used to remove nuclei bits [36].

Furthermore, a local double ellipse descriptor incorporated melanocyte characteristics that provide melanocyte recognition parameters by using 30 histopathological skin

objects with different features, such as a slice. This methodology showed a sensitivity rate of approximately 80% and a successful prediction rate of roughly 70% [37]. Li et al. [38] presented a segmentation method based on the neuro-fuzzy model. Segmentation is achieved by operating as parameters with some functions. This approach provides good accuracy, robust quality, and accuracy [39].

### 3. Proposed Method

The proposed method joins many phases and strategies to improve segmentation outcomes' accuracy and robustness. The segmentation processes will be performed to detect and segment the lesion parts from the skin automatically. To segment the image, the following are the details:

- (1) Converting color images into gray images
- (2) Creates Gaussian filters using

$$h_g(n_1, n_2) = e^{-\frac{(n_1^2 + n_2^2)}{2\sigma^2}}, \quad (1)$$

$$h(n_1, n_2) = \frac{h_g(n_1, n_2)}{\sum_{n_1} \sum_{n_2} h_g}, \quad (2)$$

equations (1) and (2),  $n_1$  and  $n_2$  are the pixel values of the image,  $\sigma$  is the standard deviation, and  $h$  is the value of the Gaussian filter.

- (3) Calculate threshold by using maximum entropy calculated by the following mathematical equations: equation (3),  $h(i)$  is a normalized histogram value, which takes integer values from 0 to 255. Hence, assume that  $h(i)$  is a normalized value, Eq. (4) shows the entropy of white pixels, Eq. (5) shows entropy of black pixels, and Eq. (6) maximizes white and black pixels' optimal threshold which can be selected:

$$\sum_{i=0}^{i_{max}} h(i) = 1, \quad (3)$$

$$H_B(t) = - \sum_{i=0}^i \frac{h(i)}{\sum_{j=0}^t h(j)} \log \frac{h(i)}{\sum_{j=0}^t h(j)}, \quad (4)$$

$$H_B(t) = - \sum_{i=i+1}^i \frac{h(i)}{\sum_{j=i+1}^t h(j)} \log \frac{h(i)}{\sum_{j=i+1}^t h(j)}, \quad (5)$$

$$T = \text{Arg}_{i=0 \dots i_{max}} \text{Max } H_b(t) + H_w(t). \quad (6)$$

- (4) Apply median filter to reduce noise and preserve edges.
- (5) To remove tiny parts from images after thresholding, all connected components or objects with fewer pixels from the binary image need to prune. Thus, the operation has been performed using eight connectivity components known as an area opening.

- (6) Morphology operations opening and closing have been applied.
- (7) Adaptive contour method is based upon the optimal mask to segment the skin lesion.
- (8) Refinement of segmentation using morphology operations has been performed.
- (9) Remove small holes outside the body.
- (10) Select one big area that shows a lesion.

This paper uses the adaptive contour (snack) segmentation method. The initialization mask is the most important for active contouring methods to feed the seed point required in that algorithm. One way is to select this mask manually for seed point, but it is not easy to apply to all types of images. Thus, it is required to propose a method shown in Figure 3 for the selection of masks automatically using the active contouring method. We have proposed a technique based on the Laplacian Gaussian filter-based maximum entropy for mask selection. Details of these steps are explained below.

According to many previous studies and research, there are many methods to perform the threshold process. After testing those methods in the actual experiments, they may provide accurate results in some cases, but they failed in many of them due to the different nature of the images. The main issue the existing algorithms faced was the low quality of the picture. Noise is sometimes present in the form of hair or shadow, which affects their results. Those methods must either choose the initial threshold value, which is a big problem, or use a calculated average weight as an initial threshold. Using average value may fail in low-quality/noisy images since the average is not robust to noise. So, we used Laplacian Gaussian filter-based maximum entropy technique that gives a dynamic and the optimal threshold.

This stage aims to provide a dynamic, optimal, and adaptive threshold. That threshold will be used in the advanced stage to perform the segmentation of the image.

Morphological operations have been used to detect the best seed point and for a clear map representing the lesion in the image. It uses the functions of morphology like opening and closing operations and the structure/synthesized operations to produce a better picture. Many factors affect the accuracy of that algorithm's output, like the characteristic of structuring elements and synthesized modes of the functions. In more detail, it can consider the synthesized model of operators reflects the relationship between the processing image and the original one and choosing the synthesized mode will affect the accuracy and the outcome. Hence, the secrets of morphological operations will be simplified for the morphological filter structure design and the selection of structuring elements. To detect the border on the medical images, we have to choose the appropriate structure element by the texture features of the picture. Factors like the shape, direction of structuring elements, and size should be considered.

In most cases, the structuring element is selected to be  $3 \times 3$  square. Dilation, erosion, opening, and closing are primary operations and functions of binary morphology. We chose the disk-structuring element using the empirical

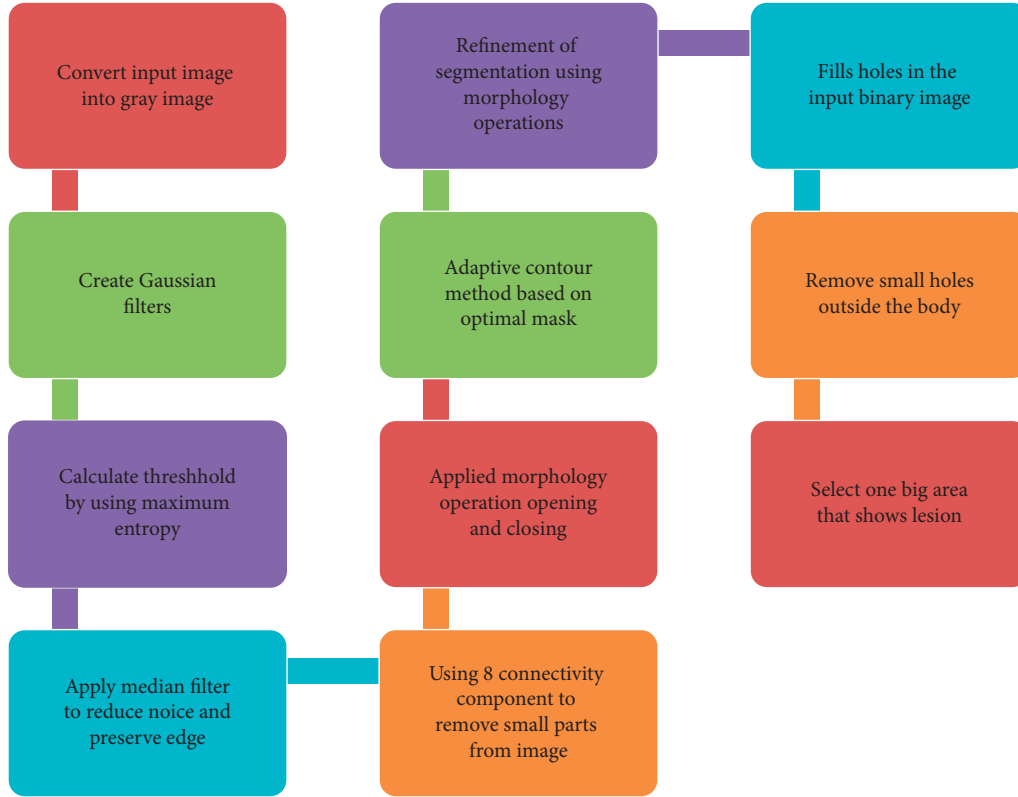


FIGURE 3: Proposed method.

method, the size of the structuring element used for opening is nine, and for closing, it is 25.

Kass introduced the central concept and the idea of the active contours “Snakes: Active Contour Model.” The definition of the snake is a dynamic curve based on the parameter, which attempted to move into a minimum energy position. The snake energy can be calculated using the energy function shown in (7) and (8), which Kass introduced.

$$E_{\text{snake}}^* = \int_0^1 E_{\text{snake}}(v(s))ds, \quad (7)$$

$$\begin{aligned} E_{\text{snake}}^* &= \int_0^1 E_{\text{snake}} \\ &= \int_0^1 E_{\text{int}}(v(s)) + E_{\text{ext}}(v(s))ds \\ &= \int_0^1 E_{\text{int}}(v(s)) + E_{\text{img}}(v(s)) + E_{\text{con}}(v(s))ds. \end{aligned} \quad (8)$$

The energy of the snake segmentation algorithm consists of three essential terms. “ $E_{\text{int}}$ ” denotes the snake’s internal power, “ $E_{\text{img}}$ ” represents the forces of the image, and “ $E_{\text{con}}$ ” rises to external constraint forces. The external snake forces ( $E_{\text{ext}}$ ) can be calculated by summing  $E_{\text{img}}$  and  $E_{\text{con}}$  values. The parametric curve “ $C$ ” is the classical snake used in this research. It is attracted to the area with intense gradients where the typical gradient weight of any point is high. The principle of snake segmentation is to

locate the initial contour in the image, which is distorted under various energies. There is a need for two kinds of energy, i.e., internal and external.

**Internal energy:** It depends on the parametric curve’s first and second derivatives representing the snake. This energy depends on the main properties of the curve and the sum of bending and elastic energy. (9) shows the internal energy equation.

$$E_{\text{int}} = E_{\text{elastic}} + E_{\text{bending}} = \int_0^1 \frac{1}{2} (\alpha |v_s|^2 + \beta |v_{ss}|^2) ds. \quad (9)$$

The external energy relates to image characteristics, such as the availability of noise or edges in the image. It guarantees that the snake is on the edge when maximizing the standard gradient’s amount over the curve and thus minimizing its opposite. External energy ( $E_{\text{ext}}$ ) of the contour comes from the image and takes on its smaller parts of the function of, for example, boundaries. (10) shows the external energy equation.

$$E_{\text{ext}} = \int_0^1 E_{\text{image}}(v(s))ds. \quad (10)$$

First, the initial contour is in the center of the circle in the image. Then, the iteration of the algorithm will move; this will cause the total energy to be reduced. A function  $E$  will be computed for every neighborhood point ( $n$ ) of active contour energy point ( $p$ ). The point  $p_0$  attempts to minimize the energy  $E_0$  and then exchange contour point  $P$  if  $E > E_0$ .

TABLE 1: Summary of existing segmentation methods.

Reference	Approach	Disadvantages
Stoecker et al. [15]	Gray-level co-occurrence matrix for texture feature	The presence of artifacts like shining areas and shadows caused by light makes the process of segmentation of skin lesion images more complicated.
Stoecker and Scharcanski [22]	Four different algorithms	To identify the region of nuclei which used the intensity and size of nuclei as a parameter
Sonali and Kamat [23]	Combined thresholding with fuzzy C-means	It may not perform well over images with huge variations in skin colors
Manju Bharathi and Sarswati [24]	NC ratio analysis for automatic segmentation of cells	Performance degrades over lesions of varying sizes and shapes
Jeniva and Santhi [25]	Learning model of natural skin texture and cancer textures	A lot of difference between specific kinds of cancer and the surrounding area of skin
Kumar et al. [26]	Local region recursive segmentation, K-means clustering, and local double ellipse descriptor	It may not perform well over images with huge variations in skin colors
Abbadi and Miry [27]	Thresholding and Wiener filter	Low lesion-to-skin gradient, depigmentation, multiple tumor regions
Lu et al. [28]	Mean shift, local region recursive segmentation, and local double ellipse descriptor	This method is computationally complex
Baral, Gonnade, and Verma [29]	Neuro-fuzzy model and some other features	Complex thresholding approaches

Otherwise, it will keep it the same. This operation will be repeated until convergence when the contour gained at iteration  $t$  equals that gained at iteration  $t + 1$ .

After applying the adaptive contour method, sometimes it produces more than one segment inside each other. Thus, it is essential to fill the big-segmented part. We have used eight connected neighborhoods to fill the larger segment.

#### 4. Results and Discussion

We have evaluated the performance of our proposed technique using a dataset of 69 dermoscopic images. These images have been gathered from DermIS database. These images belong to the two primary melanoma and non-melanoma classes. The dataset we have used contains 43 images of melanoma and 26 images of nonmelanoma. The result of the proposed segmentation is shown in Table 1. The original image column shows the input image from DermIS dataset. The second column, segmented images, shows the output after applying the segmentation method. The boundary detection column shows the detection of the boundary of the lesion in that image.

Table 2 shows the original and segmented images after applying the proposed segmentation technique. Some morphology operations have been applied to correct the images and select the mask for the adaptive contour method. We have used the adaptive contour method for segmentation. The adaptive contour method requires one mask as a seed point to start for segmentation, and then it grows with every iteration until the stopping condition meets. First, Gaussian filter-based maximum entropy has been used to calculate the adaptive and optimal threshold. Based upon this threshold, the image has been segmented, and then morphology operations opening and closing have been applied to select a specific mask for the adaptive contour method.

Thus, this mask gives a seed point to start the adaptive method. Column 1 shows the original images. Column 2 shows the segmented binary image after applying the proposed method for segmentation. In the third column, the segmented part is highlighted in green color in the original image. These images clearly show that the proposed method works for all images perfectly. The primary reason behind this good segmentation is finding the optimal mask for active contour. Segmentation results are outstanding and effective if the extracted mask is good.

In the previous phase, the segmentation was performed on dermoscopic images to extract the portion of the lesion. The proposed technique is described in the last section in detail. In this section, only the experimental results are shown.













The segmentation performance is measured and validated by the area base correlation criteria and Dice similarity coefficient frequently used in the literature. These criteria are given below.

*4.1. Jaccard Similarity Index for Region Coincidence.* The performance of the segmentation method is measured by the Jaccard similarity index based on region coincidence criteria [35, 36]. The following formula calculates the accuracy of the segmentation as in

$$P(R, A) = \frac{|A \cap R|}{|A \cup R|}, \quad 0 \leq P(R, A) \leq 1, \quad (11)$$

where  $A$  is the ground truth structure, the  $R$  segment is the detected foreground structure, and  $A$  represents the segment area computing operation. Factor  $|A \cup R|$  shows how much ground truth structure is detected. The factor  $|A \cap R|$  shows the normalization, which normalizes the accuracy measure. The value 1 indicates the ideal similarity or matching between the computed area by the system and ground truth.

TABLE 2: Original images, segmented, and boundary detected.

Original image	Segmented images	Boundary detection
		
		
		
		

**4.2. Dice Similarity Coefficient.** Dice coefficient (Dice, 1945) measures the spatial overlap of two sets. In the proposed system, this method is used to validate the performance of the segmentation method. In our case, the one set is the segmentation by the proposed method, and the other group is the ground truth segmentation by the expert. The Dice similarity coefficient is defined as in

$$DS C_{(A,R)} = \frac{2|A \cap R|}{|A| + |R|} \quad (12)$$

$A$  indicates the pixels of segmentation extracted by the proposed system, and  $R$  is the area by the expert as a ground truth. The value of DSC ranges from 0 to 1. The ideal weight of DSC is 1; i.e., the performance of the segmentation method agrees if DSC values are closer to 1.

Tables 3 and 4 show our proposed methodology result performance. The performance of the segmentation process

is frequently measured and validated by the region base coincidence criteria and Dice similarity coefficient. The value 1 indicates the ideal similarity or matching between the computed area by the system and ground truth. This approach provides good accuracy and is robust. The segmentation part can be used for feature extraction to classify skin lesions..

Comparison has been made with different existing algorithms used for dermoscopic images. Table 5 shows that the proposed method performs well using all performance measures. Dice coefficient (Dice, 1945) measures the spatial overlap of two sets and shows 0.969. Similarly, Jaccard similarity index for region coincidence also shows the best result 0.969. Both parameters show the best result. Our method has higher segmentation performance and capability compared with previous method.

TABLE 3: Accuracy measure of segmentation with ground truth from melanoma images.

Dataset	Data size (pixels)	$ A \cap R $ (Pixels)	$ A \cup R $ (Pixels)	P (out of 1)	DSC (out of 1)
Image# 6	226 × 276	835	845	0.967	0.963
Image# 7	226 × 210	1794	1707	0.955	0.954
Image# 8	226 × 276	1543	1569	0.934	0.931
Image# 9	226 × 210	1181	1189	0.924	0.952
Image# 10	224 × 275	1303	1321	0.944	0.931
Image# 12	224 × 275	2154	2171	0.931	0.923
Image# 14	226 × 275	1769	1784	0.947	0.951
Image# 15	226 × 276	2711	2743	0.941	0.942
Image# 16	224 × 276	3049	3064	0.934	0.932
Image# 27	224 × 210	1628	1643	0.942	0.943

TABLE 4: The accuracy measure of segmentation with ground truth from nonmelanoma images.

Dataset	Data size (pixels)	$ A \cap R $ (Pixels)	$ A \cup R $ (Pixels)	P (out of 1)	DSC (out of 1)
Img44	226 × 276	10023	10103	0.945	0.944
Img45	226 × 210	14324	14379	0.963	0.969
Img46	226 × 276	3145	3304	0.952	0.958
Img47	226 × 210	3823	4011	0.943	0.951
Img48	224 × 275	359	367	0.927	0.923
Img49	224 × 275	2768	2731	0.946	0.941
Img50	226 × 275	466	456	0.933	0.939
Img51	226 × 276	4167	4107	0.925	0.927
Img52	224 × 276	2572	2519	0.921	0.928
Img53	224 × 210	933	956	0.938	0.932

TABLE 5: Comparing results with existing methods.

Method	Nonmelanoma images		Melanoma images	
	DSC (out of 1)	P (out of 1)	DSC (out of 1)	P (out of 1)
Long et al. [40]	90.46	82.59	89.03	80.22
Badrinarayanan et al. [41]	91.32	84.03	9 84.55	73.23
Ronneberger et al. [42]	89.88	81.63	82.04	69.55
Al-Masni et al. [43]	91.38	84.13	92.92	86.77
Proposed method	0.969	0.963	0.963	0.967

## 5. Conclusion and Future Work

This research presents an automated and novel set of approaching techniques for the segmentation of dermoscopic images using automated and intelligent methods. Proper segmentation of skin cancer melanoma and subsequent diagnoses using automatic computer-aided design (CAD) systems is not only a trending research phenomenon but also a necessity of modern times. This objective was met by developing techniques and combining them within one framework, which not only promised efficiency. But, it also achieved better results than the existing solutions to this problem.

This research proposes a new skin lesion segmentation method by combining Gaussian filter-based maximum entropy-based threshold for mask selection, morphology operations, and adaptive contour method. We use Jaccard similarity index for region coincidence and Dice similarity coefficient to evaluate this method. Our method shows better results when

comparing it with the image segmented by the expert dermatologist. Thus, the major contribution is to propose a mask selection for active contour and morphology operations as a preprocessing and postprocessing of active contour.

In the future, if we will combine preprocessing and postprocessing stages which extract the features and classify the input skin cancer images into melanoma and non-melanoma using a machine learning classifier, our modal performance will be better on all medical application.

### Data Availability

The data used in this research will be available upon request from the corresponding author.

### Conflicts of Interest

The authors declare that they have no conflicts of interest.



## Acknowledgments

The authors would like to thank the support of the Deanship of Scientific Research at Princess Nourah bint Abdulrahman University. This research project was funded by Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R239), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.





## References

- [1] B. Sreedhar, M. S. Be, and M. S. Kumar, "A comparative study for classification of skin cancer," in *Proceedings of the 2019 International Conference on System Science and Engineering (ICSSE)*, pp. 267–272, Dong Hoi, Vietnam, July 2019.
- [2] M. Kamran, M. Malik, M. W. Iqbal, M. Anwar, M. Aqeel, and S. Ahmad, "Web simplification prototype for cognitive disabled users," *Human Behavior and Emerging Technologies*, vol. 2022, no. 1, pp. 1–14, 2022.
- [3] I. Filali, M. Belkadi, R. Aoudjit, and M. Lalam, "Graph weighting scheme for skin lesion segmentation in macroscopic images," *Biomedical Signal Processing and Control*, vol. 68, no. 1, p. 102710, 2021.
- [4] J. K. Winkler, C. Fink, F. Toberer et al., "Association between surgical skin markings in dermoscopic images and diagnostic performance of a deep learning convolutional neural network for melanoma recognition," *JAMA Dermatology*, vol. 155, no. 10, p. 1135, 2019.
- [5] T. Naz, M. Akhtar, S. K. Shahzad, M. Fasli, M. W. Iqbal, and M. R. Naqvi, "Ontology-driven advanced drug-drug interaction," *Computers & Electrical Engineering*, vol. 86, no. 1, p. 106695, 2020.
- [6] K. Liu, M. Mokhtari, B. Li, S. Nofallah, and C. May, "Learning melanocytic proliferation segmentation in histopathology images from imperfect annotations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3766–3775, IEEE, Nashville, Tennessee, 19–25 June 2021.
- [7] R. B. Oliveira, A. S. Pereira, and J. M. R. S. Tavares, "Computational diagnosis of skin lesions from dermoscopic images using combined features," *Neural Computing & Applications*, vol. 31, no. 10, pp. 6091–6111, 2019.
- [8] L. R. Araújo, R. D. Andrade, P. C. Rodrigues, and R. V. Silva, "Automatic segmentation of melanoma skin cancer using deep learning," in *Proceedings of the 2020 IEEE International Conference on E-health Networking, Application & Services (HEALTHCOM)*, pp. 1–6, IEEE, Shenzhen, China, March 2021.
- [9] S. Albahli, H. T. Rauf, M. Arif, M. T. Nafis, and A. Algosaihi, "Identification of thoracic diseases by exploiting deep neural networks," *Neural Networks*, vol. 5, no. 6, p. 45, 2021.
- [10] D. Didona, G. Paolino, U. Bottoni, and C. Cantisani, "Non melanoma skin cancer pathogenesis overview," *Biomedicines*, vol. 6, no. 1, p. 6, 2018.
- [11] M. Arif, F. Ajesh, S. Shamsudheen, O. Geman, D. Izdrui, and D. Vicoveanu, "Brain tumor detection and classification by MRI using biologically inspired orthogonal wavelet transform and deep learning techniques," *Journal of Healthcare Engineering*, vol. 4, pp. 1–18, 2022.
- [12] S. Bechelli and J. Delhommelle, "Machine learning and deep learning algorithms for skin cancer classification from dermoscopic images," *Bioengineering*, vol. 9, no. 3, p. 97, 2022.
- [13] M. Arif, G. Wang, V. E. Balas, and S. Chen, "Band segmentation and detection of dna by using fast fuzzy c-mean and neuro adaptive fuzzy inference system," in *International conference on smart city and informatization*, vol. 23, pp. 49–59, Springer, Singapore, 2019, November.
- [14] M. Arif, G. Wang, O. Geman, and J. Chen, "Medical image segmentation by combining adaptive artificial bee colony and wavelet packet decomposition," in *Proceedings of the International Conference on Dependability in Sensor, Cloud, and Big Data Systems and Applications*, vol. 12, pp. 158–169, Singapore, November 2019.
- [15] W. V. Stoecker, C. S. Chiang, R. H. Moss, and R. H. Moss, "Texture in skin images: comparison of three methods to determine smoothness," *Computerized Medical Imaging and Graphics*, vol. 16, no. 3, pp. 179–190, 1992.
- [16] O. Geman, I. Chiuchisan, I. Ungurean, M. Hagan, and M. Arif, "Ubiquitous healthcare system based on the sensors network and android internet of things gateway," in *Proceedings of the 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCOM/IOP/SCI)*, pp. 1390–1395, IEEE, Guangzhou, China, October 2018.
- [17] J. Azmi, M. Arif, M. T. Nafis, M. A. Alam, S. Tanweer, and G. Wang, "A systematic review on machine learning approaches for cardiovascular disease prediction using medical big data," *Medical Engineering & Physics*, vol. 105, p. 103825, 2022.
- [18] M. Nawaz, Z. Mehmood, T. Nazir et al., "Skin cancer detection from dermoscopic images using deep learning and fuzzy k-means clustering," *Microscopy Research and Technique*, vol. 85, no. 1, pp. 339–351, 2022.
- [19] U. Jamil, S. Khalid, M. U. Akram, A. Ahmad, and S. Jabbar, "Melanocytic and nevus lesion detection from diseased dermoscopic images using fuzzy and wavelet techniques," *Soft Computing*, vol. 22, no. 5, pp. 1577–1593, 2018.
- [20] M. Malik, M. Waseem Iqbal, S. Khuram Shahzad et al., "Determination of COVID-19 patients using machine learning algorithms," *Intelligent Automation & Soft Computing*, vol. 31, no. 1, pp. 207–222, 2022.
- [21] S. Öztürk and U. Özkaya, "Skin lesion segmentation with improved convolutional neural network," *Journal of Digital Imaging*, vol. 33, no. 4, pp. 958–970, 2020.
- [22] P. G. Cavalcanti and J. Scharcanski, "Automated prescreening of pigmented skin lesions using standard cameras," *Computerized Medical Imaging and Graphics*, vol. 35, no. 6, pp. 481–491, 2011.
- [23] S. R. Jadhav and D. K. Kamat, *Segmentation Based Detection of Skin Cancer*, IRF International Conference, 2014.
- [24] S. Saraswathi, "Cancer cell segmentation and detection using NC ratio analysis," *International Journal of Renewable Energy Technology*, vol. 03, no. 19, pp. 549–552, 2014.
- [25] S. Jeniva and C. Santhi, "An efficient skin lesion segmentation analysis using statistical texture distinctiveness," *Algorithms*, vol. 32, p. 21, 2015.
- [26] R. Kumar, P. Kumar, R. Tripathi, G. P. Gupta, A. K. M. N. Islam, and M. Shorfuzzaman, "Permissioned blockchain and deep-learning for secure and efficient data sharing in industrial healthcare systems," *IEEE Transactions on Industrial Informatics*, p. 1, 2022.
- [27] N. El Abbadi and A. H. Miry, "Automatic segmentation of skin lesions using histogram thresholding," *Journal of*

- Computer Science*, vol. 10, no. 4, pp. 632–639, Article ID 1549-3636, 2014.
- [28] C. Lu, M. Mahmood, N. Jha, and M. Mandal, “Automated segmentation of the melanocytes in skin histopathological images,” *IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS*, vol. 17, no. 2, pp. 284–296, MARCH 2013.
- [29] B. Baral, S. Gonnade, and T. Verma, “Lesion segmentation in dermoscopic images using decision based neuro fuzzy model,” *International Journal of Computer Science and Information Technologies*, vol. 5, no. 2, pp. 2546–2552, 2014.
- [30] S. Khoulood, M. Ahlem, T. Fadel, and S. Amel, “W-net and inception residual network for skin lesion segmentation and classification,” *Applied Intelligence*, vol. 52, no. 4, pp. 3976–3994, 2022.
- [31] M. A. Marchetti, N. C. Codella, S. W. Dusza et al., “Results of the 2016 international skin imaging collaboration international symposium on biomedical imaging challenge: comparison of the accuracy of computer algorithms to dermatologists for the diagnosis of melanoma from dermoscopic images,” *Journal of the American Academy of Dermatology*, vol. 78, no. 2, pp. 270–277.e1, 2018.
- [32] N. C. F. Codella, Q. B. Nguyen, S. Pankanti et al., “Deep learning ensembles for melanoma recognition in dermoscopy images,” *IBM Journal of Research and Development*, vol. 61, no. 4/5, pp. 1–5, 2017.
- [33] A. Esteva, B. Kuprel, R. A. Novoa et al., “Dermatologist-level classification of skin cancer with deep neural networks,” *Nature*, vol. 542, no. 7639, pp. 115–118, 2017.
- [34] J. Zhang, Y. Xie, Q. Wu, and Y. Xia, “Medical image classification using synergic deep learning,” *Medical Image Analysis*, vol. 54, no. 1, pp. 10–19, 2019.
- [35] I. Gonzalez-Diaz, “Dermaknet: incorporating the knowledge of dermatologists to convolutional neural networks for skin lesion diagnosis,” *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 2, pp. 547–559, 2019.
- [36] A. Kensert, P. J. Harrison, and O. Spjuth, “Transfer learning with deep convolutional neural networks for classifying cellular morphological changes,” *SLAS Discovery*, vol. 24, no. 4, pp. 466–475, 2019.
- [37] N. Liu, X. Lu, L. Wan, H. Huo, and T. Fang, “Improving the separability of deep features with discriminative convolution filters for RSI classification,” *ISPRS International Journal of Geo-Information*, vol. 7, no. 3, p. 95, 2018.
- [38] Z. Li, X. Zhang, H. Müller, and S. Zhang, “Large-scale retrieval for medical image analytics: a comprehensive review,” *Medical Image Analysis*, vol. 43, no. no, pp. 66–84, 2018.
- [39] T. Shakeel, S. Habib, and W. Boulila, “A survey on COVID-19 impact in the healthcare domain: worldwide market implementation, applications, security and privacy issues, challenges and future prospects,” *Complex Intelligent Syst*, vol. 3, p. 34, 2022.
- [40] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, April 2017.
- [41] V. Badrinarayanan, A. Handa, and R. Cipolla, “Segnet: a deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling,” arXiv preprint arXiv:1505.07293, 2015.
- [42] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, Cham, 2015.
- [43] M. A. Al-Masni, M. A. Al-antari, M. T. Choi, S. M. Han, and T. S. Kim, “Skin lesion segmentation in dermoscopy images via deep full resolution convolutional networks,” *Computer Methods and Programs in Biomedicine*, vol. 162, pp. 221–231, 2018.

## Research Article

# Predicting Divorce Prospect Using Ensemble Learning: Support Vector Machine, Linear Model, and Neural Network

Mian Muhammad Sadiq Fareed <sup>1</sup>, Ali Raza,<sup>2</sup> Na Zhao <sup>3</sup>, Aqil Tariq <sup>4</sup>, Faizan Younas,<sup>2</sup> Gulnaz Ahmed,<sup>2</sup> Saleem Ullah,<sup>2</sup> Syeda Fizzah Jillani <sup>5</sup>, Irfan Abbas,<sup>6</sup> and Muhammad Aslam<sup>7</sup>

<sup>1</sup>Department of Software Engineering, University of Central, Punjab 54000, 1-Khayaban-e-Jinnah Road, Johar Town, Lahore, Pakistan

<sup>2</sup>Department of Computer Science, Khwaja Fareed University of Engineering and Information Technology, Rahim Yar Khan 64200, Pakistan

<sup>3</sup>State Key Laboratory of Resources and Environmental Natural Resources Research, Chinese Academy of Sciences, Beijing 100101, China

<sup>4</sup>State Key Laboratory of Information Engineering in Surveying Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China

<sup>5</sup>Department of Physics, Physical Sciences Building, Aberystwyth University, Aberystwyth SY23, UK

<sup>6</sup>School of Agricultural Equipment Engineering, Jiangsu University, Zhenjiang 212013, China

<sup>7</sup>School of Computing Engineering and Physical Sciences, University of West of Scotland, Paisley, UK

Correspondence should be addressed to Na Zhao; [zhaon@reis.ac.cn](mailto:zhaon@reis.ac.cn)

Received 17 March 2022; Revised 20 April 2022; Accepted 23 May 2022; Published 11 July 2022

Academic Editor: Le Sun

Copyright © 2022 Mian Muhammad Sadiq Fareed et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A divorce is a legal step taken by married people to end their marriage. It occurs after a couple decides to no longer live together as husband and wife. Globally, the divorce rate has more than doubled from 1970 until 2008, with divorces per 1,000 married people rising from 2.6 to 5.5. Divorce occurs at a rate of 16.9 per 1,000 married women. According to the experts, over half of all marriages ends in divorce or separation in the United States. A novel ensemble learning technique based on advanced machine learning algorithms is proposed in this study. The support vector machine (SVM), passive aggressive classifier, and neural network (MLP) are applied in the context of divorce prediction. A question-based dataset is created by the field specialist. The responses to the questions provide important information about whether a marriage is likely to turn into divorce in the future. The cross-validation is applied in 5 folds, and the performance results of the evaluation metrics are examined. The accuracy score is 100%, and Receiver Operating Characteristic (ROC) curve accuracy score, recall score, the precision score, and the F1 accuracy score are close to 97% confidently. Our findings examined the key indicators for divorce and the factors that are most significant when predicting the divorce.

## 1. Introduction

Divorce (or dissolution of marriage) is the definitive termination of a marital partnership, canceling the legal duties and responsibilities of marriage and dissolving the parties' matrimonial relations. In other terms, divorce is a constitutional action taken by married people to end their marriage. It is also known as marriage dissolution and is the constitutional step that ends a marriage ahead when either partner dies.

In general, there are two sorts of divorce. One option is "divorce from bed and board," which is legal in some jurisdictions. At its essence, this permits couples to legally separate and is frequently utilized by spouses who want to live their own lives but do not want to formally break their marriage for whatever reason. Divorce from bed and board is uncommon these days. An "absolute divorce," which terminates the marriage, is the most prevalent kind of divorce, therefore, to speak, a legal clean break. This topic will be the subject of this

article. There are several ways to achieve the aim of having a court issue, an absolute divorce ruling. For convenience, it has been the usual practice in law to classify each of these procedures as a different type of divorce, which we will do here.

The following states contain divorce data for the United States. There have been 2,015,603 weddings. Marriage occurs at a rate of 6.1 per 1,000 of the population in total. There have been 746,971 divorces. Divorce occurs at a rate of 2.7 per 1,000 people (45 reporting states) [1].

Divorce occurs at a rate of 16.9 per 1,000 married women. Many experts believe that this is a far more authentic representation of the genuine divorce rate [2] than the raw number. The divorce rate for every 1,000 married women is about double of what it was in 1960; nonetheless, it is lower than the all-time high of 22.6 in the early 1980s. In the United States, about half of the total marriages end in separation or divorce. According to the researchers, 41% of all the first marriages result in divorce. The second marriages fail about 60%. All third marriages end in divorce about 73%. The United States has the world's sixth highest divorce rate [3].

Machine learning is an artificial intelligence (AI) technique that enables computers to automatically develop and learn on their own without being explicitly programmed. Machine learning [4] is anxious with the establishment of computer programmers that can access information data and employ it to learn on their own. Text classification [5] is a machine learning approach that assigns tags or categories to text automatically. Text classifiers can evaluate and categorize text by sentiment [6], subject, and consumer intent using natural language processing (NLP) [7] quicker and more correctly than people.

Ensemble modeling is an effective method for improving the performance of our model. It typically pays to use ensemble learning in addition to any other models we may be developing. Ensemble learning techniques [8] are a kind of machine learning methodology that accommodates numerous base techniques to create the best prediction technique.

The divorce prospect prediction is the core objective of this novel research study. The main contributions of this research are the following:

- (i) A novel research study in terms of divorce prospect prediction using a questionnaire dataset is proposed in this paper.
- (ii) The three advanced machine learning models, support vector machine (SVM), passive aggressive classifier (PAC), and neural networks (multilayer perceptron classifier) are utilized for the prediction task. Our employed techniques are fully hyperparameter tuned.
- (iii) An enhanced novel ensemble learning approach based on three machine learning techniques is employed to predict the divorce prospect of the couple.
- (iv) The divorce exploratory data analysis (DEDA) is conducted to get fruitful insights to form the dataset and to determine the major factors that cause divorce.
- (v) The cross-validation (CV) is applied in 5 folds, and the performance results evaluation metric of the proposed approach is examined.

- (vi) The comparative analysis of model performance is conducted among the three employed SVM, PAC, and Neural network approaches.

The rest of the paper is formulated as: The divorce-related work is examined in Section 2. The architectural methodology analysis of our proposed research approach is analyzed in Section 3. The applied advanced machine learning techniques are examined in Section 4. Then, a novel ensemble learning approach based on three machine learning techniques is discussed in Section 5. The results and evaluation of the proposed approaches are explained and deliberated in Section 6. Then, to conclude the research work, Section 7 contains the conclusion of this novel research study.

## 2. Related Work

The authors used Yöntem's findings to construct 56 questions as divorce predictors. Furthermore, they employed four automated learning models (perceptron, logistic regression, neural networks, and randomized forest) as well as three hybrid models based on voting criteria. Each of these models was trained in 5 distinct scenarios, resulting in a total of 35 tests, with the performance attained in terms of accuracy, sensitivity, and specificity is 0.98, 1.0, and 0.96, respectively, for the perceptron model and a hybrid model [9].

The categorization approaches are used to forecast divorce in Turkey. In 2019, the authors carried out this investigation. They determined in this study that the ANN technique paired with a correlation-based matrix of feature space selection performs best, with an accuracy score of 98% and a Kappa value of 0.97. The SVM model training span is also less than that of the ANN model training span [10].

The authors utilized significant characteristics in this suggested study by deleting duplicate features that do not help with the prediction by applying an improved machine learning technique to the standard dataset accessible to forecast the divorce rate. They were able to reach 99% accuracy. This technique may also be utilized as evidence by family counseling professionals on a couple's emotional and psychological well-being [11].

Within the area of this study, divorce prediction was performed utilizing the Divorce Predictors Scale based on the Gottman couple's therapy. DPS's success was explored utilizing the multilayer perceptron (MLP) neural networks and decision tree algorithms. The study also seeks to identify the most important features of the Divorce Predictor Scale values that influence divorce. When the direct classification learning methods were applied to the divorce dataset, the RBF neural network had the greatest success rate of 98%. This scale can be used by family counselors and family therapists to help with the case formulation and intervention planning. Furthermore, the predictors of divorce in the Gottman couple relation therapy were verified in the Turkish samples [12].

In a long-term, prospective longitudinal research, this paper explores the predictability of divorce. During the 14-year research period, the prediction was attainable with a technique that incorporated marital happiness, concerns of

the marriage breakup, and emotional interaction in both talks. The algorithm correctly predicted divorce 93% of the time [13].

An artificial neural network (ANN) technique was created and employed in this research to predict whether or not a couple will divorce. The prediction is based on several questions that the couple acknowledged, and the answers to those questions served as the input data to the ANN model. The model was subjected to repeated learning over training data and validation cycles until it achieved 100% accuracy [14].

The authors are offering a study on the prediction of divorce cases using available machine learning techniques in this paper. The authors compared the accuracy of the perceptron learning classifier, random forest learning classifier, decision tree learning classifier, Naive Bayes learning classifier, support vector machine learning classifier, and K-nearest neighbor learning classifier for divorce case prediction. Following training, the algorithm will forecast whether or not the divorce will materialize. This allows the therapist to assess how stressful a couple's condition is and properly counsel them. With the perceptron model, the authors attained an accuracy of 98% [15].

The detection of COVID-19 based on a blood test was proposed in this study [16]. The ensemble-learning-based approach was developed for the prediction of COVID-19. At the first stage of research, the deep-learning-based classifier convolutional neural network (CNN) was utilized. The dataset was used from the San Raffaele Hospital. In the second stage of research, the 15 different machine-learning-based classifiers were applied. The findings of the research study show that the ensemble learning model achieved an accuracy score of 99%.

Malware detection based on ensemble learning techniques is proposed in this study [17]. The fully connected convolutional neural network (CNN)-based classifier was developed for base stage classification. The machine-learning-based models were utilized for end-stage classification. 15 machine-learning-based classifiers were utilized for malware detection. The dataset of Windows Portable Executable (PE) malware was used for model training and testing results. The research findings show that the fully connect CNN ensemble model and machine-learning-based extra trees classifier achieved an accuracy score of 100%.

In conclusion, our proposed novel research study is based on the prediction of divorce prospects using ensemble learning techniques. The comparative analysis with the past applied research study shows that our research study outperformed by utilizing advanced techniques. The research study results' outcomes are efficient, validated, and higher than the past applied approaches. We have revealed the key indicators for divorce and the factors that are the most significant when predicting divorce in this research study.

### 3. Methodology

The methodological analysis of the proposed research study is analyzed in this section. The working flow of our research findings flow is elaborated here.

The questionnaire dataset is analyzed and useful insights are taken from it. Feature engineering is applied to make a

predictable model with the best-fit features in the context of divorce prediction. The data normalization is applied to make the dataset in perfect form for our proposed model.

Now dataset splitting is applied to split the dataset into two portions. The 80% portion of the data is used for model training and 20% is utilized for model testing and performance evaluation. The three models are applied with the ensemble learning approach. Finally, the ensemble learning model prediction is used for predicting the divorce.

The research methodology for this novel research is examined in Figure 1. It visualizes the workflow of the complete research study. In the first step, the questionnaire dataset is analyzed by the exploratory data analysis (EDA). Then, in the next step, feature engineering is applied to get the useful features for the ensemble learning model. Then, the data normalization is applied. The dataset splitting is applied in the next step. Then, the train portion is given to the model, and then, the test model results in the evaluation of the test portion. After all these methodology steps are done, a predictive ensemble learning model is formed and ready to predict the divorce of a couple.

**3.1. Dataset.** The dataset is based on the questions asked by the specialists to the married couples [18]. The answers to these 54 questions will predict the chance of divorce between them. The questions are graded on a scale of 0 to 4, with 0 being the worst and 4 being the best. The last category indicates whether or not the couple has divorced. Table 1 contains the descriptive dataset analysis.

**3.2. Divorce Exploratory Data Analysis.** The divorce exploratory data analysis (DEDA) refers to the essential process of administrating preliminary investigations on data to spot anomalies. The uncovered data patterns can be found by applying DEDA. The test hypotheses are performed using DEDA. The assumption validation using graphical representations and summary statistics is demonstrated by utilizing the DEDA.

The bar plot is a plot on the Divorce\_Y\_N column in Figure 2. In the bar plot, 0 represents the number of divorce class and 1 represents the divorce class. The bar plot shows the total number of divorces and not divorce value. The value of divorce in Figure 2 is 86, and the value of number of divorce is 84. The bar chart shows that the data set is balanced. Both classes have approximately the same number of rows.

The violin chart is the plot based on the dataset to explore the cause of divorce in Figures 3–5. A violin graph is a cross between a kernel density plot and a box plot that visualizes the data peaks. It is utilized to display how numerical data points are distributed in the employed dataset.

As opposite to a box plot, which can only bring summary statistics, violin graphs visualize summary statistics as well as the frequency of every variable. In the violin plot of the I'm\_not\_wrong (51) column, we explore that as the intensity of value increases, the number of divorces increases, and as the value decreases, the number of divorces decreases. The

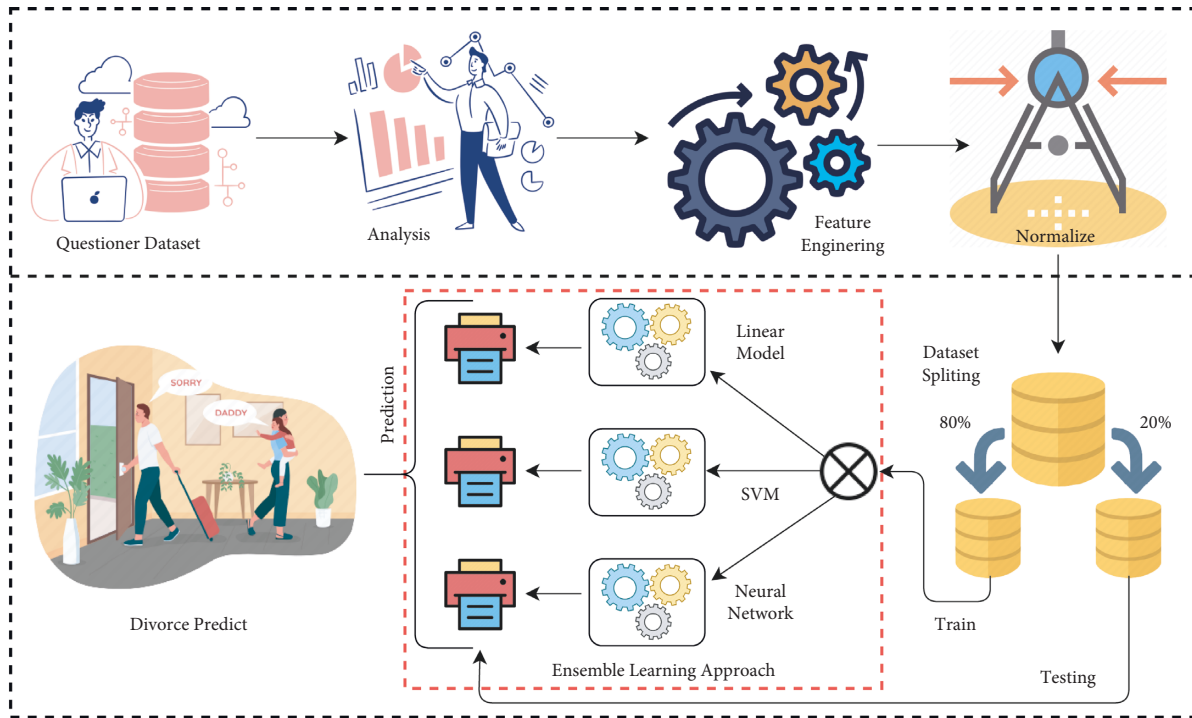


FIGURE 1: The methodology diagram of the proposed research system.

analysis graph also shows that it has a great impact on the Divorce\_Y\_N column.

In Figure 3, the data from the violin plot is also explored with the column of love (16), common goal (10), and enjoy holidays (8). The graph shows the cause of divorce and no\_divorce when the value of the scale changes. The violin plot is also plotted on the column of happy (17), always never (32), trust (21), and you are inadequate (53) in Figure 4.

The violin plot shows how the cause of divorce changes when the scale changes. The violin plot of argue\_then\_leave (42), humiliate (36), and friend social (30) is analyzed in Figure 5. In Figure 5, we explored whether the effect of divorce change is linked with the scale change through the violin plot..

All these applied divorce analyses prove to be very fruitful in the context of getting useful insights from the dataset and its related features.

The histogram chart is the plot of the dataset in Figures 6 and 7. A histogram is referred to as a data representation tool, which appears to be a bar chart that buckets a variation of outcomes along with the x-axis columns. The numerical value count or percent of value occurrences in the dataset for every column is represented on the y-axis.

We get the histogram of features 2\_stranger (7), silence\_instead\_of\_discussion (45), I'm\_not\_wrong (51), good\_to\_leave\_home (44), I'm\_not\_guilty (50), humiliate (36), not\_calm (37), negative\_personality (33), and know\_well (29) and get the total number of counts in the different scale values. The histogram is the plot of insult (35), common\_goal (10), no\_home\_time (6), special\_time (5), contact (4), begin\_correct (3), ignore\_diff (2), incompetence (54), always\_never (32), and by counting the number of different scale values.

The histogram is the plot of the features friends\_social (30), know\_well (29), hopes\_wishes (28), current\_stress

(27), anxieties (26), inner\_world (25), fav\_food (23), care\_sack (22), and likes (21) showing the total number of counts on the y-axis and the 0 to 4 scale on the x-axis. The histogram chart is plotted on trust, role, marriage, love, and dreams columns and explored the number of counts on a different scale on the y-axis and x-axis, respectively.

From Figure 6, we have analyzed that the feature I'am\_not\_wrong (51) has higher rank values among all. This shows that this feature question has a major cause of divorce and that's why it has higher ranked scale values.

This applied divorce histogram analysis is based on the prominent questions present in the dataset and their scale ranks. These questions are analyzed to get their feature importance and to determine the relationship between divorce causes. These features are for model training and getting divorce prediction from it.

A correlation graph displays the correlations for various variables present in the dataset employed. The correlation matrix emphasizes the relationship between all the possible pairings of values in a dataset. It is a powerful tool for summarizing a large dataset in addition to visualizing and identifying trends in the provided data. We draw the correlation matrix on the dataset in Figure 8. The visualized features are based on the correlation values above or equal to 0.7. The feature that has low correlation values is not present in the feature display map. The correlation matrix shows that all features are highly related. All features are important to use for the training of our model.

**3.3. Feature Engineering.** The technique of changing the raw dataset into a prominent feature space that well describes the root problem of predictive techniques, resulting in improving

TABLE 1: The dataset attribute details.

Question no.	Question by the specialist
1	If one of us apologizes when our discussion deteriorates, the discussion ends.
2	I know we can ignore our differences, even if things get hard sometimes.
3	When we need it, we can take our discussions with my spouse from the beginning and correct them.
4	When I discuss this with my spouse, contacting him will eventually work.
5	The time I spent with my wife is special for us.
6	We don't have time at home as partners.
7	We are like two strangers who share the same environment at home rather than family.
8	I enjoy our holidays with my wife.
9	I enjoy traveling with my wife.
10	Most of our goals are common to my spouse.
11	I think that one day in the future when I look back, I see that my spouse and I have been in harmony with each other.
12	My spouse and I have similar values in terms of personal freedom.
13	My spouse and I have a similar sense of entertainment.
14	Most of our goals for people (children, friends, etc.) are the same.
15	Our dreams with my spouse are similar and harmonious.
16	We're compatible with my spouse about what love should be.
17	We share the same views about being happy in our life with my spouse.
18	My spouse and I have similar ideas about how marriage should be.
19	My spouse and I have similar ideas about how roles should be in marriage.
20	My spouse and I have similar values in trust.
21	I know exactly what my wife likes.
22	I know how my spouse wants to be taken care of when she/he is sick.
23	I know my spouse's favorite food.
24	I can tell you what kind of stress my spouse is facing in her/his life.
25	I know my spouse's inner world.
26	I know my spouse's basic anxiety.
27	I know what my spouse's current sources of stress are.
28	I know my spouse's hopes and wishes.
29	I know my spouse very well.
30	I know my spouse's friends and their social relationships.
31	I feel aggressive when I argue with my spouse.
32	When discussing with my spouse, I usually use expressions such as "you always" or "you never."
33	I can use negative statements about my spouse's personality during our discussions.
34	I can use offensive expressions during our discussion.
35	I can insult my spouse during our discussion.
36	It can be humiliating when we have discussions.
37	My discussion with my spouse is not calm.
38	I hate my spouse's way of opening a subject.
39	Our discussions often occur suddenly.
40	We're just starting a discussion before I know what's going on.
41	When I talk to my spouse about something, my calm suddenly breaks.
42	When I argue with my spouse, I only go out and I do not say a word.
43	I mostly stay silent to calm the environment a little.
44	Sometimes I think it's good for me to leave home for a while.
45	I'd rather stay silent than discuss it with my spouse.
46	Even if I'm right in the discussion, I stay silent to hurt my spouse.
47	When I discuss this with my spouse, I stay silent because I am afraid of not being able to control my anger.
48	I feel right in our discussions.
49	I have nothing to do with what I have been accused of.
50	I'm not the one who's guilty of what I am accused of.
51	I'm not the one who's wrong about problems at home.
52	I wouldn't hesitate to tell my spouse about her/his inadequacy.
53	When I discuss, I remind my spouse of her/his inadequacy.
54	I'm not afraid to tell my spouse about her/his incompetence.

the employed model accuracy results on the unseen dataset, is referred to as the feature engineering technique. The 54 features of the divorce questionnaire dataset are used as dependent features, and the target feature containing the label class is utilized in this research study. The top 10

absolute correlation features are examined in Figure 9. The fav\_food (24), know\_well (30), freedom\_value (12), marriage (18), special\_time (5), roles (19), harmony (11), happy (17), enjoy\_travel (9), insult (36), humiliate (37), and trust (21) are the top correlated features.

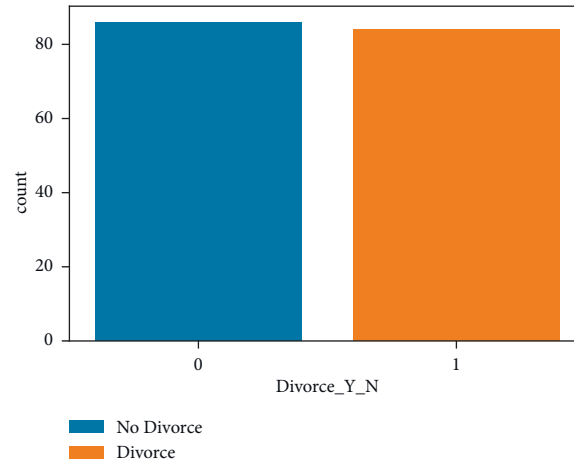


FIGURE 2: The divorce dataset balancing analysis by the target class.

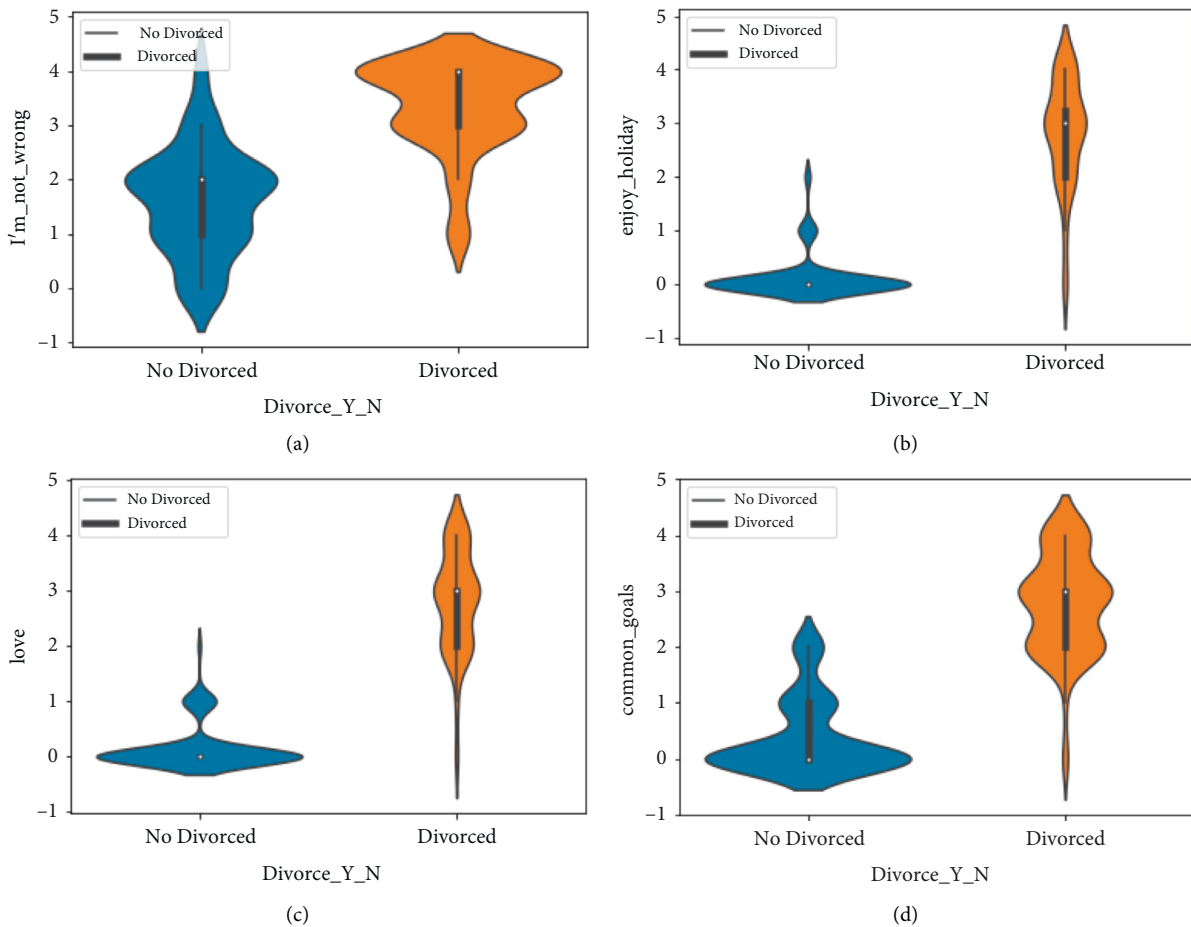


FIGURE 3: Divorce analysis by I'm\_not\_wrong, enjoy\_holiday, love, and common\_goals features. (a) The violin graph analysis of I'm\_not\_wrong feature among Divorced and not Divorced category, (b) The violin graph analysis of enjoy\_holiday feature among Divorced and not Divorced category, (c) The violin graph analysis of love feature among Divorced and not Divorced category, and (d) The violin graph analysis of common\_goals feature among Divorced and not Divorced category.

**3.4. Dataset Splitting.** Dataset splitting appears as a requirement for removing bias from training data in machine learning systems. The dataset is split into two sets: the training dataset, which is used by the model to learn an efficient mapping of inputs to output, and the test set, which

is utilized to effectively assess the proposed model's result performance. This division prevents the employed technique from overfitting [19]. The dataset splitting utilized in this research has a ratio of 80: 20. The 80% portion of the dataset is used to ensemble learning models, and the 20% portion of



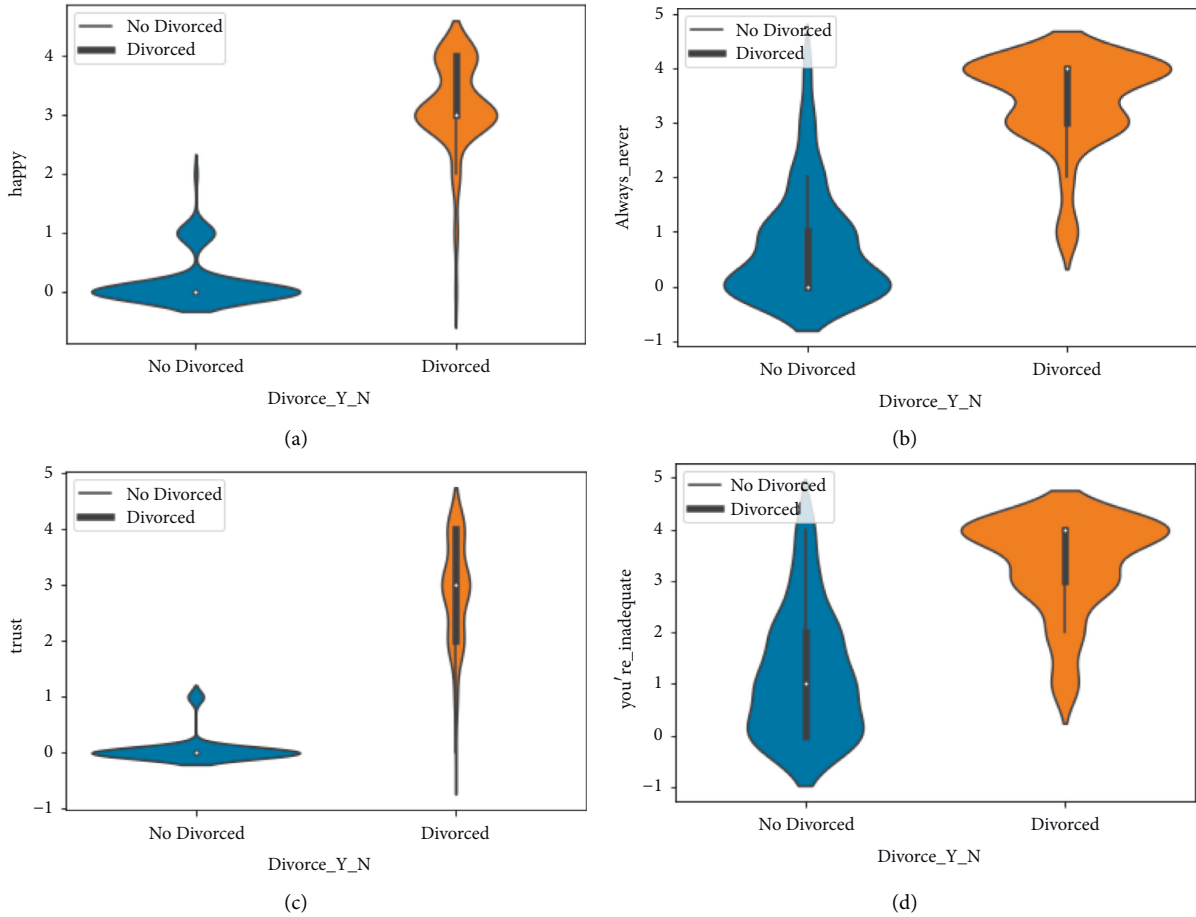


FIGURE 4: Divorce analysis by happy, always\_never, trust, and you're\_inadequate features. (a) The violin graph analysis of happy feature among Divorced and not Divorced category, (b) The violin graph analysis of always\_never feature among Divorced and not Divorced category, (c) The violin graph analysis of trust feature among Divorced and not Divorced category, and (d) The violin graph analysis of you're\_inadequate feature among Divorced and not Divorced category.

the dataset is utilized for testing and evaluating the ensemble model. The random state unit for splitting is 42.

## 4. Proposed Approaches

**4.1. Passive Aggressive Classifier.** The passive-aggressive categorization [20] is one of the accessible incremental learning methods because it uses a closed-form updating rule. In the sense that they do not require a learning rate, passive-aggressive algorithms are akin to perceptron models. They do, however, contain a regularization parameter. The classifier updates its weight vector for each misclassified training sample it gets in an attempt to fix it. The hyperparameters by tuning analysis of the passive-aggressive algorithm are examined in Table 2.

**4.2. Support Vector Machine.** The support vector machine (SVM) [21] is a supervised learning model that is utilized to solve regression and classification problems. It is largely employed in categorization-related difficulties. Every data item is visualized as a point in  $n$ -dimensional space, where  $n$  is the number of data features. The value of every data feature

is the worth of a certain coordinate in the SVM model. Then, we achieve classification by establishing the hyper-plane that best distinguishes the two classes of the employed dataset. The SVM technique hyperparameters are analyzed in Table 3.

**4.3. Neural Networks.** A feedforward artificial neural network (ANN) that generates a set of outputs from a set of employed inputs is referred to as a multilayer perceptron (MLP) neural network [22]. An MLP is referred to by various layers of employed input nodes that are associated as a directed graph between the output and input layers. Backpropagation is utilized by MLP to train the employed neural network. An MLP is a neural network that joins many layers in a directed graph, which means that the data signal routed across the graph nodes is only a single direction. In addition to the input nodes, every node has an activation function of the nonlinear form.

Backpropagation [23] is a supervised machine learning technique utilized by an MLP. The MLP is a deep-learning-based approach since it uses various layers of neurons. The

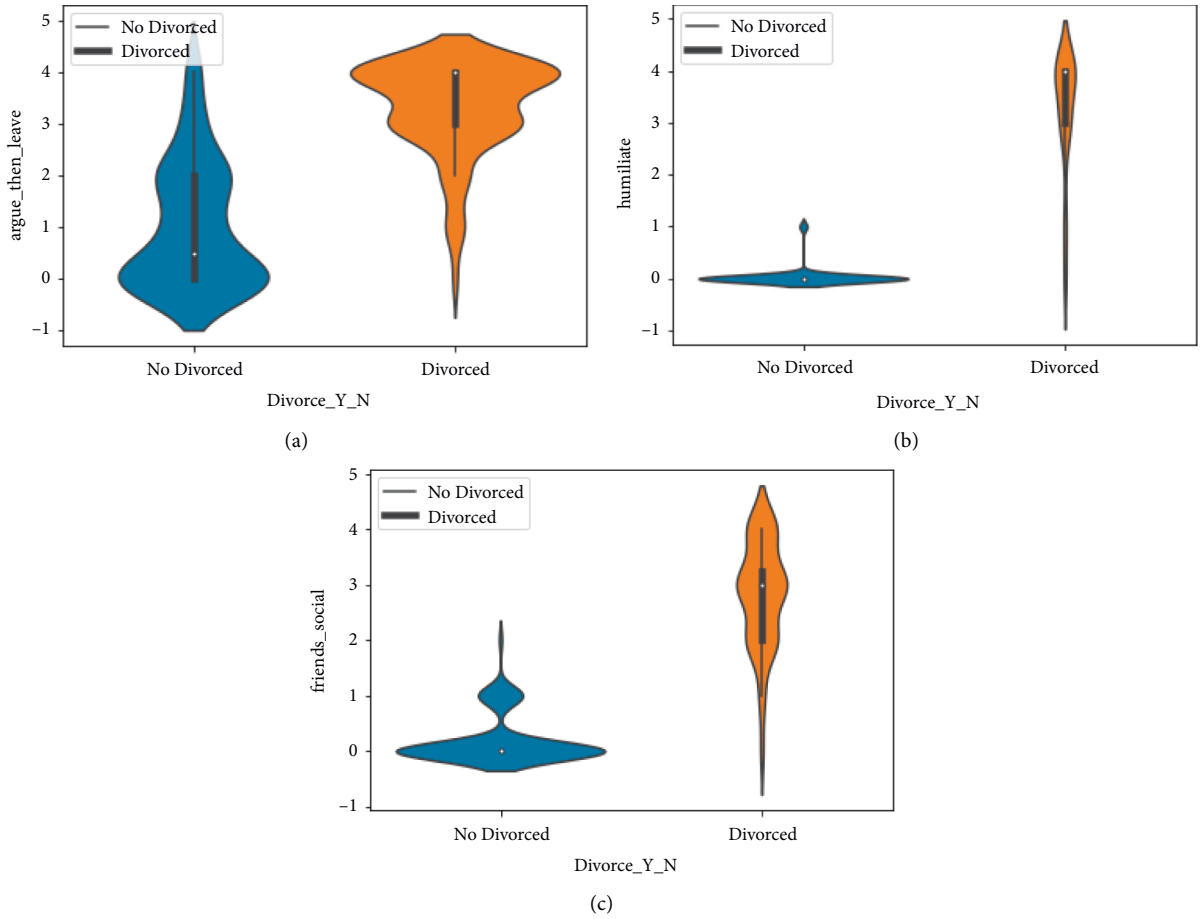


FIGURE 5: The divorce analysis by argue\_then\_leave, humiliates, and friends\_social features. (a) The violin graph analysis of argue\_then\_leave feature among Divorced and not Divorced category, (b) The violin graph analysis of humiliates feature among Divorced and not Divorced category, (c) The violin graph analysis of friends\_social feature among Divorced and not Divorced category.

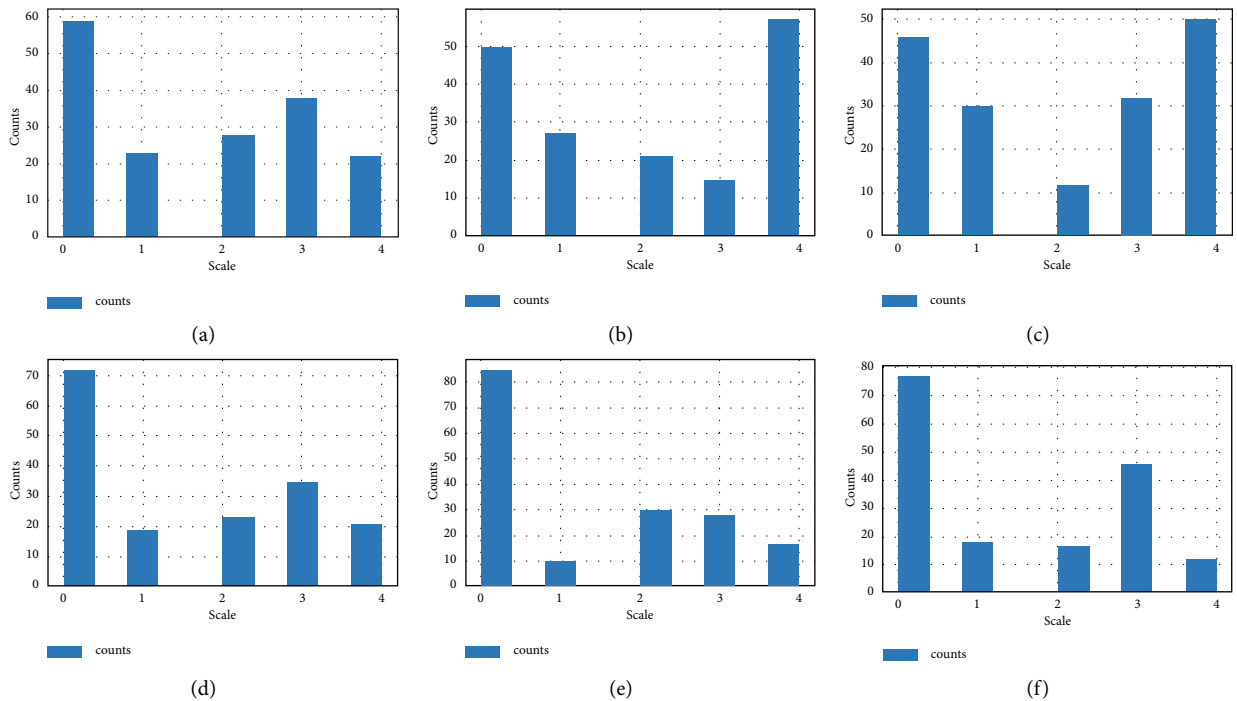


FIGURE 6: Continued.

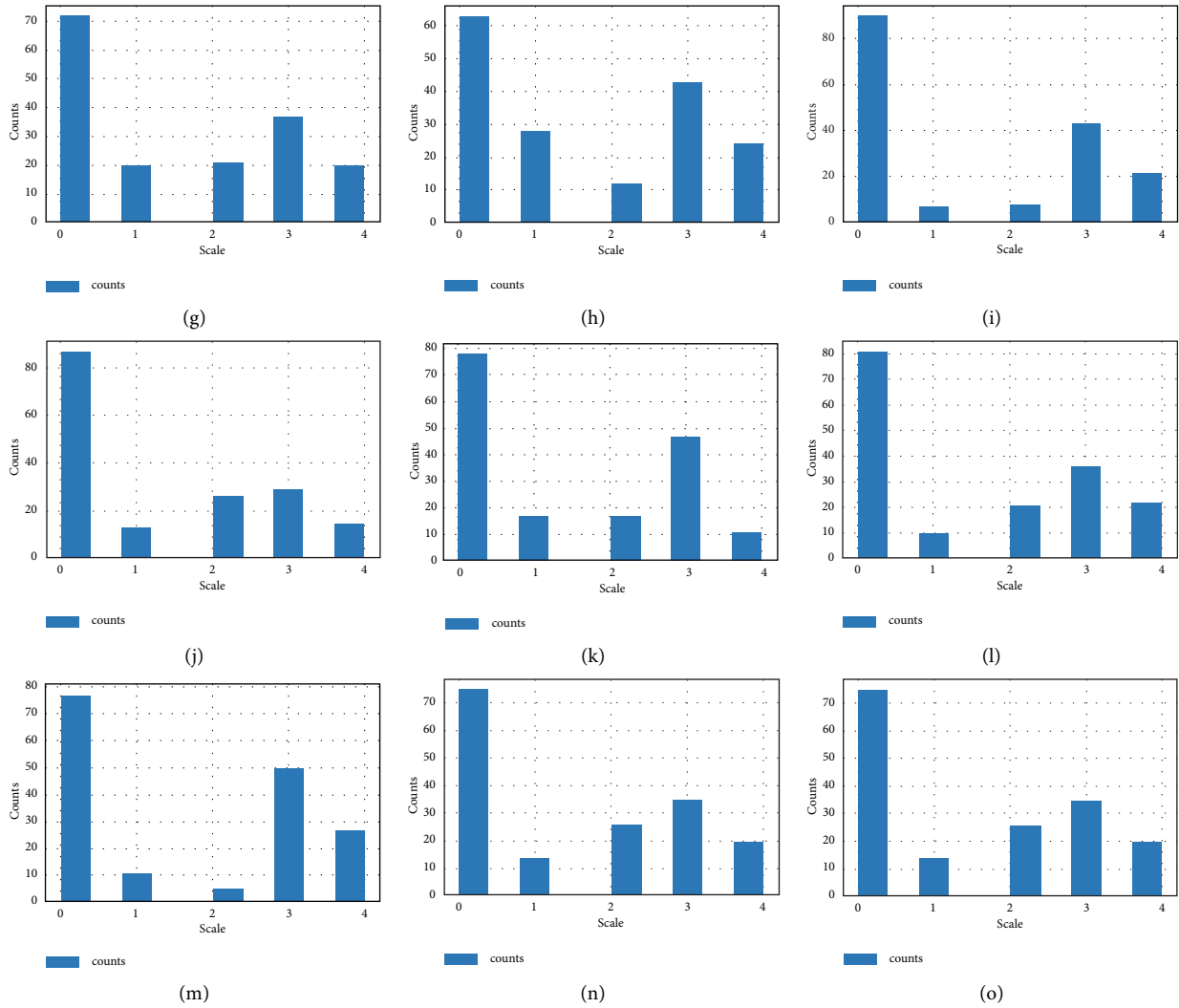


FIGURE 6: The divorce histogram analysis of 15 prominent questions scale ranks analysis. (a) The ranked scale analysis being the lowest and highest for feature Ignore\_diff, (b) The ranked scale analysis being the lowest and highest for feature incompetence, (c) The ranked scale analysis being the lowest and highest for feature Always\_never, (d) The ranked scale analysis being the lowest and highest for feature friends\_social, (e) The ranked scale analysis being the lowest and highest for feature hopes\_wishes, (f) The ranked scale analysis being the lowest and highest for feature current\_stress, (g) The ranked scale analysis being the lowest and highest for feature anxieties, (h) The ranked scale analysis being the lowest and highest for feature inner\_world, (i) The ranked scale analysis being the lowest and highest for feature fav\_food, (j) The ranked scale analysis being the lowest and highest for feature care\_sick, (k) The ranked scale analysis being the lowest and highest for feature likes, (l) The ranked scale analysis being the lowest and highest for feature trust, (m) The ranked scale analysis being the lowest and highest for feature roles, (n) The ranked scale analysis being the lowest and highest for feature marriage and (o) The ranked scale analysis being the lowest and highest for feature love.

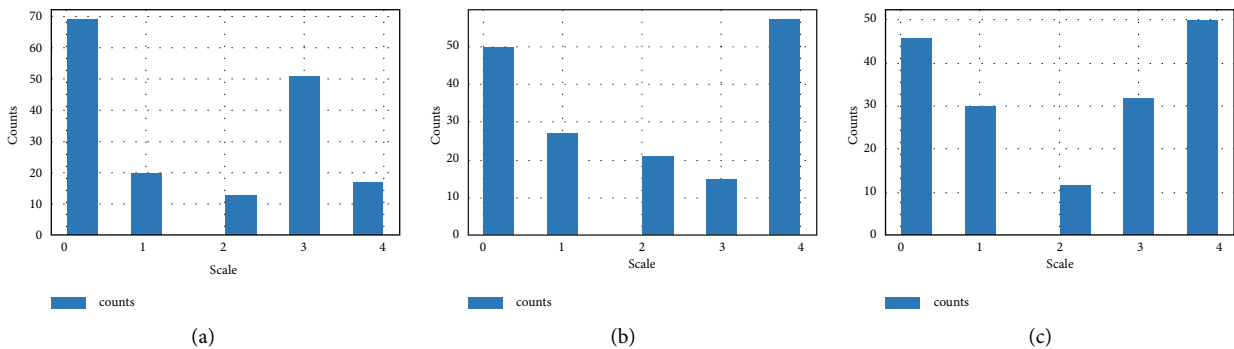


FIGURE 7: Continued.

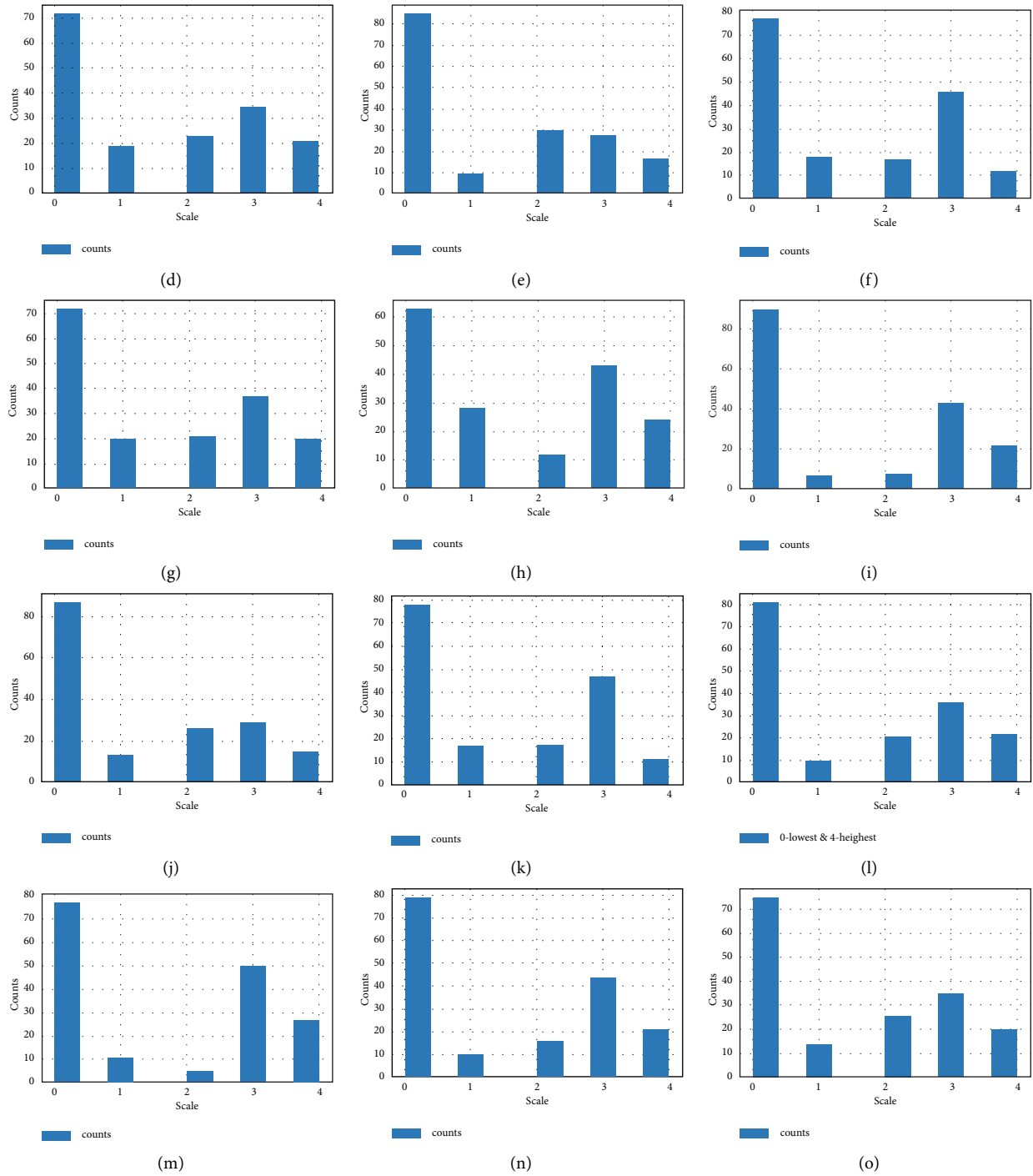


FIGURE 7: The divorce histogram analysis of other 15 prominent questions scale ranks analysis. (a) The ranked scale analysis being the lowest and highest for feature dreams, (b) The ranked scale analysis being the lowest and highest for feature incompetence, (c) The ranked scale analysis being the lowest and highest for feature Always\_never, (d) The ranked scale analysis being the lowest and highest for feature friends\_social, (e) The ranked scale analysis being the lowest and highest for feature hopes\_wishes, (f) The ranked scale analysis being the lowest and highest for feature current\_stress, (g) The ranked scale analysis being the lowest and highest for feature anxieties, (h) The ranked scale analysis being the lowest and highest for feature inner\_world, (i) The ranked scale analysis being the lowest and highest for feature fav\_food, (j) The ranked scale analysis being the lowest and highest for feature care\_sick, (k) The ranked scale analysis being the lowest and highest for feature likes, (l) The ranked scale analysis being the lowest and highest for feature trust, (m) The ranked scale analysis being the lowest and highest for feature roles, (n) The ranked scale analysis being the lowest and highest for feature marriage and (o) The ranked scale analysis being the lowest and highest for feature love.

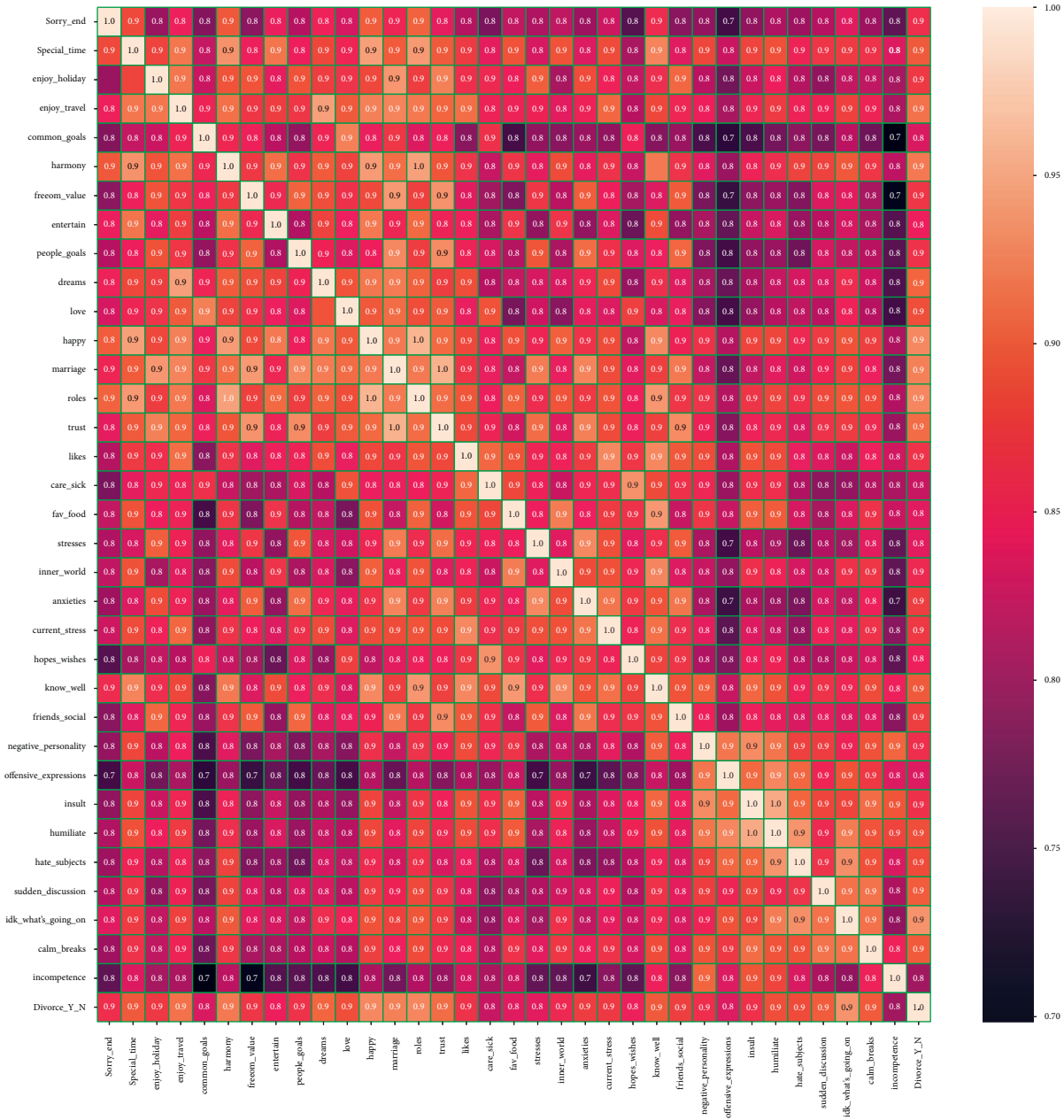


FIGURE 8: Dataset correlation analysis.

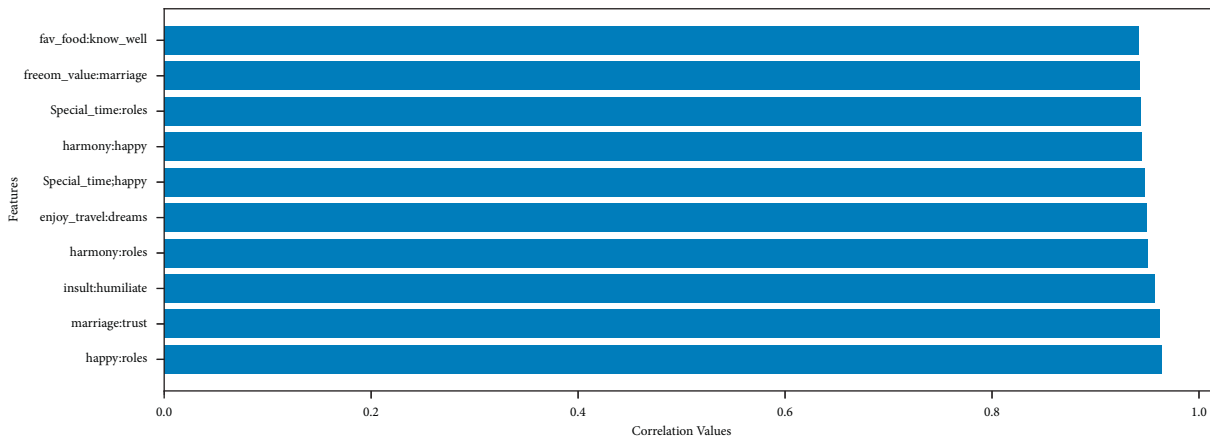


FIGURE 9: The top 10 absolute correlation feature analyses.

TABLE 2: The applied model hyperparameters by tuning.

Proposed technique	Hyperparameters		
	Max iterations	Verbose	Random state
Passive aggressive classifier (PAC)	300	0	50

TABLE 3: The applied model hyperparameters by tuning.

Proposed technique	Hyperparameters		
	Max iterations	Kernel	Random state
Support vector machine (SVM)	300	Linear	10

TABLE 4: The applied model hyperparameters by tuning.

Proposed technique	Hyperparameters					
	Hidden layers	Activation	Random state	Verbose	Max iterations	Solver
Neural networks (MLP)	200	Logistic	50	0	200	Adam

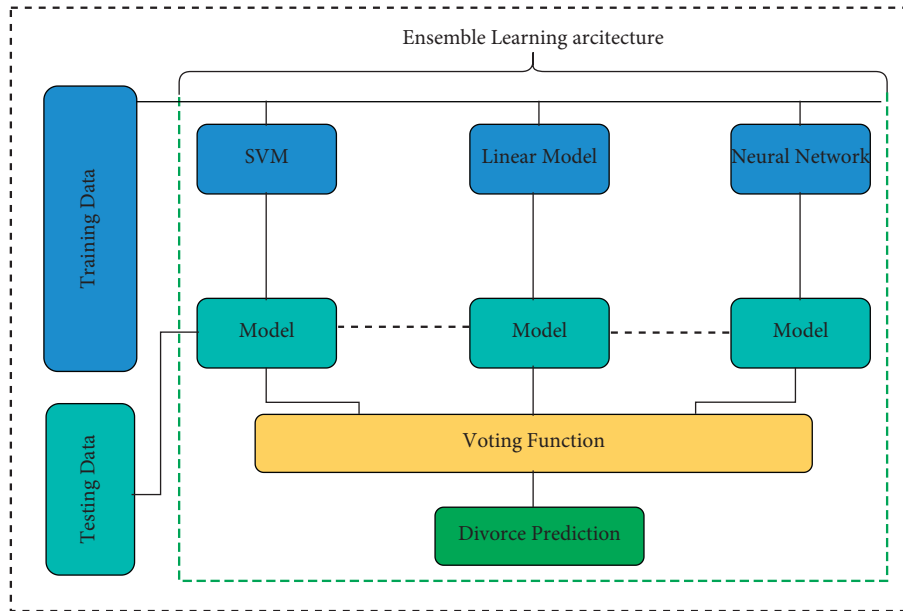


FIGURE 10: The proposed ensemble learning architecture analysis.

TABLE 5: The comparison analysis of selected methods before and after hyperparameter tuning.

Proposed technique	Before hyperparameter tuning		After hyperparameter tuning	
	Accuracy score	Training time (seconds)	Accuracy score	Training time (seconds)
Support vector machine (SVM)	97	0.004660367965698242	100	0.0017824172973632812
Passive aggressive classifier (PAC)	97	0.0012810230255126953	97	0.002166748046875
Neural network (MLP)	97	0.9576735496520996	100	0.4841580390930176

MLP is mostly utilized for supervised learning tasks, in addition to research into parallel distributed computing and computational neuroscience. Speech recognition, machine translation, and picture recognition are some of the applications of MLP. The hyperparameters analysis of MLP is examined in Table 4.

TABLE 6: The  $k$ -fold cross-validation results of applied machine learning approaches.

Sr. no.	Proposed technique	Accuracy score %
1	Support vector machine (SVM)	98
2	Passive aggressive classifier (PAC)	98
3	Neural network (MLP)	98

TABLE 7: The comparative analysis of the proposed ensemble techniques.

Proposed technique	Comparative analysis metrics		
	Accuracy %	Log loss	Training time (seconds)
Support vector machine (SVM)	100	$9.992007221626415e - 16$	0.0017824172973632812
Passive aggressive classifier (PAC)	97	1.0158463645561975	0.002166748046875s
Neural network (MLP)	100	$9.992007221626415e - 16$	0.4841580390930176
Ensemble learning (EL)	100	$9.992007221626415e - 16$	1.0685508251190186

TABLE 8: The ensemble learning performance evaluation results.

Proposed technique	Performance evaluation metrics					
	Accuracy %	ROC accuracy %	Precision accuracy %	Recall accuracy %	F1 score %	Log loss
Ensemble learning (EL)	100	97	97	97	97	$9.992007221626415e - 16$

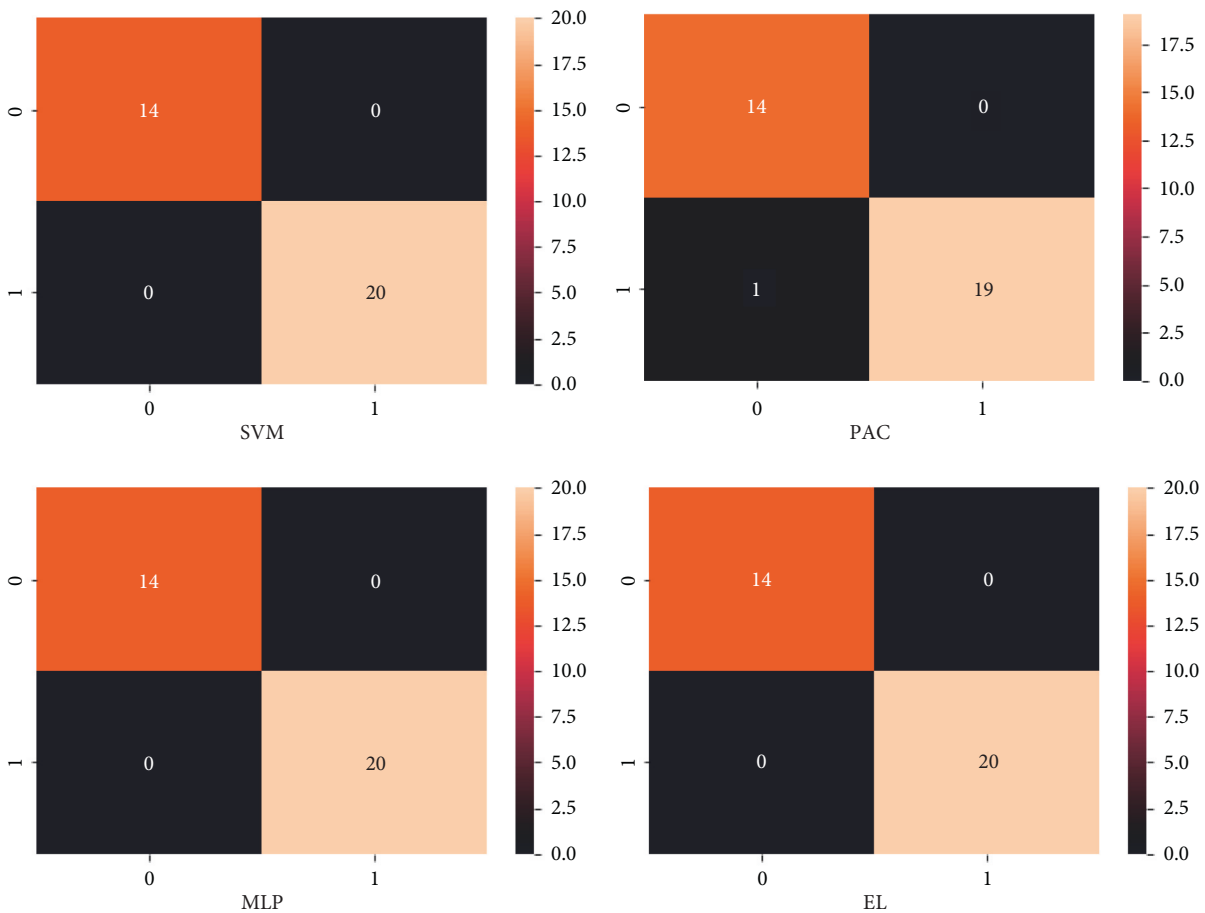


FIGURE 11: The proposed ensemble learning approach confusion matrix.

### 5. Ensemble Learning

The ensemble learning approach is examined and applied in this research. The architecture of the applied approach, the ensemble approach, is analyzed in Figure 10. The training dataset is used for training the three classification models utilized in this research. The SVM, linear model, and neural network model are trained and tested parallelly using the pipeline of ensemble learning. The ensemble learning architecture is based on the logic to train and test all model underlying models in parallel. Now, the testing results are used by the “hard” voting function

to find the average accuracy of the model. We have applied hard voting because our classification data depends on class labels and the associated weights with every classifier. The higher accuracy score is our best prediction value.

### 6. Results and Evaluation

All performance evaluation metrics utilized in this research are examined in this section. The ensemble learning model accuracy score value, ROC accuracy score value, recall score value,

precision score value, and F1 score values are the performance evaluation metrics employed in this research study. One parameter for assessing the classification models is accuracy. The accuracy score value is the percentage of the correct number of predictions made by our proposed model. The accuracy of our proposed technique is 100%. Formally, accuracy is represented by using the following mathematical equation:

$$\text{accuracy} = \frac{\text{number of correct predictions}}{\text{total number of predictions}}. \quad (1)$$

The ROC curve is referred to as the probability curve analysis that displays the true positive rate (TPR) outcome vs the false positive rate (FPR) outcome at numerous threshold settings, separating the signal data from the noise data. The area under the curve (AUC) is a measure of an employed learning classifier's ability to discriminate between classes and is utilized to summarize the ROC curve. The ROC AUC of our proposed technique is 97%. The mathematical equation expresses the ROC AUC score:

$$\text{ROC AUC} = \int_0^1 \int_{-\infty}^{+\infty} \text{ROC}_x(t) dF_{(X|D=1)(x)} dt, \quad (2)$$

$$\text{ROC AUC} = \int_{-\infty}^{+\infty} \text{AUC}_x dF_{(x|D=1)(x)}. \quad (3)$$

Precision is referred to as the ratio of true positives rate (TPR) outcomes to all positive outcomes. The recall is a measure of how well our model identifies true positives. In our case, both have a 97% score. The mathematical equation that expressed the precision and recall:

$$\text{precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}}, \quad (4)$$

$$\text{recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}}. \quad (5)$$

The F1 score value is measured by taking the weighted average value of recall and precision. As a result, this score value examines both the false positives rate (FPR) and the false negatives rate (FNR). The F1 score is periodically more valuable than the accuracy score value, exclusively if the dataset class distribution is not equal. In our situation, the F1 score is 97%. Mathematically, it is reparented as follows:

$$F1 \text{ score} = 2 * \frac{(\text{recall} * \text{precision})}{(\text{recall} + \text{precision})}. \quad (6)$$

The hyperparameter tuning results before and after are analyzed in Table 5. The k-fold cross validation comparative results are analyzed in Table 6. The applied learning techniques comparative analysis with the ensemble learning approach is demonstrated in Tables 7 and 8.

A confusion matrix (CM) analysis is referred to as a summary of the employed classification problem and the prediction outcomes as visualized in Figure 11. The number of right and wrong predictions is summarized with count values and divided by dataset category. The CM displays

several methods in which the classification technique gets perplexed when making predictions. It is critical to assess the model's performance once it has been trained using some training data. When we developed a confusion matrix, we had several components:

- (i) Positive (P): the projected outcome is positive (like the couple gets a divorce).
- (ii) Negative (N): the projected outcome is negative (like a couple does not get a divorce).
- (iii) True positive (TP): in this case, TP denotes the expected and actual values, which are both 1 (true).
- (iv) True negative (TN): TN denotes the projected value, while 0 denotes the actual value (false).
- (v) False negative (FN): in this case, FN denotes that the predicted count value is 0 (N) while the actual count value is 1 (P). Both values in this case do not correspond. As a result, it is an FN.

## 7. Conclusion

The prediction of divorce by using machine learning and ensemble learning techniques is the core motive of this research study. The findings of our study are based on key indicators for divorce and the factors that are most significant when predicting divorce. The support vector machine (SVM), passive aggressive classifier, and neural network (MLP) are applied to predict divorce. The cross-validation and performance evaluation techniques are manipulated to evaluate the proposed models. Our EL proposed technique achieved the highest accuracy of 100%. In the context of limitations and future directions, we will try to enhance the questionnaire dataset by adding more questions to get more clarified results and also apply the data augmentation techniques. To reduce overfitting, we will explore different deep learning models.

## Data Availability

The supporting data for the findings of this study are available from the corresponding author on reasonable request.

## Consent

Informed consent was obtained from all subjects involved in the study.

## Conflicts of Interest

The authors declare no conflicts of interest.

## Authors' Contributions

M. M. S. F. conceptualized the study, carried out supervision, conducted the survey, and assisted in data collection. A. R. and F. Y. performed data analysis and manuscript writing. G. A., M. A, N. Z, and A. T. provided resources and carried



out data curation, funding acquisition, and project administration. All authors have read and agreed to the published version of the manuscript.

## Acknowledgments

This research was funded by the National Natural Science Foundation of China, grant number 42071374.

## References

- [1] "CDC. FastStats - Marriage and Divorce," 2022, <https://www.cdc.gov/nchs/fastats/marriage-divorce.htm>.
- [2] J. Dall'Agnola and H. Thibault, "Online temptations: divorce and extramarital affairs in Kazakhstan," *Religions*, vol. 12, no. 8, pp. 654–20, 2021.
- [3] E. San Diego, "Divorce Statistics and Facts | what Affects Divorce Rates in the U.S.?", 2022, <https://www.wf-lawyer-s.com/divorce-statistics-and-facts/>.
- [4] Jorge Díaz-Ramírez, "Aprendizaje Automático y Aprendizaje Profundo," *Ingeniare. Revista chilena de ingeniería*, vol. 29, no. 2, pp. 182–183, 2021.
- [5] X. Luo, "Efficient English text classification using selected machine learning techniques," *Alexandria Engineering Journal*, vol. 60, no. 3, pp. 3401–3409, 2021.
- [6] W. Chen, Z. Xu, X. Zheng, Q. Yu, and Y. Luo, "Research on sentiment classification of online travel review text," *Applied Sciences*, vol. 10, no. 15, p. 5275, 2020.
- [7] P. M. Nadkarni, L. Ohno-Machado, and W. W. Chapman, "natural language processing: an introduction," *Journal of the American Medical Informatics Association*, vol. 18, no. 5, pp. 544–551, 2011.
- [8] K. Pham, D. Kim, S. Park, and H. Choi, "Ensemble learning-based classification models for slope stability analysis," *Catena*, vol. 196, Article ID 104886, 2021.
- [9] N. Flores, S. Silva, C. Science, S. Silva, A. I. Group, and C. Science, "Machine learning model to predict the divorce OF a married couple," *3C Tecnología\_Glosas de innovación aplicadas a la pyme*, pp. 83–95, 2021.
- [10] N. Hafidz, Sfenrianto, Y. Sfenrianto, Y. Pribadi, E. Fitri, and Ratino, "ANN and SVM algorithm in divorce predictor," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 3, pp. 2523–2527, 2020.
- [11] P. Ranjitha and A. Prabhu, "Improved divorce prediction using machine learning- particle swarm optimization (PSO)," in *Proceedings of the 2020 Int. Conf. Emerg. Technol. INCET 2020*, pp. 1–5, Belgaum, India, June 2020.
- [12] Y. Mustafa Kemal, A. Kemal, İL. Tahsin, and K. Serhat, "Divorce prediction using correlation based feature selection and artificial neural networks," *Neşehir Hacı Bektaş Veli Üniversitesi SBE Derg.* vol. 9, no. 1, pp. 259–273, 2019.
- [13] J. M. Gottman and R. W. Levenson, "The timing of divorce: predicting when a couple will divorce over a 14-year period," *Journal of Marriage and Family*, vol. 62, no. 3, pp. 737–745, 2000.
- [14] I. Nasser, "Predicting whether a couple is going to get divorced or not using artificial neural networks," *Int. J. Eng. Inf. Syst.* vol. 3, no. 10, pp. 49–55, 2019.
- [15] A. Sharma, A. S. Chudhey, and M. Singh, "Divorce case prediction using machine learning algorithms," in *Proceedings of the Int. Conf. Artif. Intell. Smart Syst. ICAIS 2021*, pp. 214–219, Coimbatore, India, March 2021.
- [16] Olusola O. Abayomi-Alli, Robertas Damaševičius, Rytis Maskeliūnas, and Sanjay Misra, "An Ensemble Learning Model for COVID-19 Detection from Blood Test Samples," *Sensors*, vol. 22, no. 6, p. 2224, 2022.
- [17] N. A. Azeez, O. E. Odufuwa, S. Misra, J. Oluranti, and R. Damaševičius, "Windows PE malware detection using ensemble learning," *Informatics*, vol. 8, no. 1, 2021.
- [18] Csafrít, "Predicting divorce | kaggle," 2022, <https://www.kaggle.com/datasets/csafrít2/predicting-divorce>.
- [19] A. Rácz, D. Bajusz, and K. Héberger, "Effect of dataset size and train/test split ratios in qsar/qspr multiclass classification," *Molecules*, vol. 26, no. 4, pp. 1–16, 2021.
- [20] A. Y. Muaad, J. Hanumanthappa, M. A. Al-antari, J. V. Bibal Benifa, and C. Chola, "AI-based misogyny detection from Arabic levantine twitter tweets," in *Proceedings of the 1st Online Conf. Algorithms*, October. 2021.
- [21] A. Rizwan, N. Iqbal, R. Ahmad, and D. H. Kim, "Wr-svm model based on the margin radius approach for solving the minimum enclosing ball problem in support vector machine classification," *Applied Sciences*, vol. 11, no. 10, 2021.
- [22] A. Taravat, S. Proud, S. Peronaci, F. Del Frate, and N. Oppelt, "Multilayer perceptron neural networks model for meteosat second generation SEVIRI daytime cloud masking," *Remote Sensing*, vol. 7, no. 2, pp. 1529–1539, 2015.
- [23] N. B. Shaik, S. R. Pedapati, S. A. Ammar Taqvi, A. R. Othman, and F. A. Abd Dzubir, "A feed-forward back propagation neural network approach to predict the life condition of crude oil pipeline," *Processes*, vol. 8, no. 6, 2020.

## Research Article

# Multiclass Cancer Prediction Based on Copy Number Variation Using Deep Learning

Haleema Attique,<sup>1</sup> Sajid Shah ,<sup>1,2</sup> Saima Jabeen,<sup>3</sup> Fiaz Gul Khan,<sup>1</sup> Ahmad Khan,<sup>1</sup> and Mohammed ELAffendi <sup>2</sup>

<sup>1</sup>Department of Computer Science, COMSATS University Islamabad, Abbottabad Campus, Islamabad, Pakistan

<sup>2</sup>EIAS Data Science Lab, College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia

<sup>3</sup>Department of IT and Computer Science, Pak-Austria Fachhochschule: Institute of Applied Sciences and Technology, Mang, Haripur, KPK, Pakistan

Correspondence should be addressed to Sajid Shah; [sshah@psu.edu.sa](mailto:sshah@psu.edu.sa)

Received 31 March 2022; Accepted 21 May 2022; Published 9 June 2022

Academic Editor: Mohamed Abdelaziz

Copyright © 2022 Haleema Attique et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

DNA copy number variation (CNV) is the type of DNA variation which is associated with various human diseases. CNV ranges in size from 1 kilobase to several megabases on a chromosome. Most of the computational research for cancer classification is traditional machine learning based, which relies on handcrafted extraction and selection of features. To the best of our knowledge, the deep learning-based research also uses the step of feature extraction and selection. To understand the difference between multiple human cancers, we developed three end-to-end deep learning models, i.e., DNN (fully connected), CNN (convolution neural network), and RNN (recurrent neural network), to classify six cancer types using the CNV data of 24,174 genes. The strength of an end-to-end deep learning model lies in representation learning (automatic feature extraction). The purpose of proposing more than one model is to find which architecture among them performs better for CNV data. Our best model achieved 92% accuracy with an ROC of 0.99, and we compared the performances of our proposed models with state-of-the-art techniques. Our models have outperformed the state-of-the-art techniques in terms of accuracy, precision, and ROC. In the future, we aim to work on other types of cancers as well.

## 1. Introduction

The change in the DNA refers to the term genetic variation which makes us all unique. There are different forms of genetic variation, and most of them are well understood. It can involve changes in the DNA nucleotide or chromosome structure [1, 2]. Human genome is well-off in structural variation where copy number variation (CNV) is the most communal type which is the change in the number of copies in a specific area of the genome [3]. In the 1000 Genome Project data, CNV is known as copy number polymorphism (CNP) [4]. CNVs are DNA regions ranging in size from 1k bases to several megabases [5]. CNV is normally due to insertion, deletion, and/or duplication of the chemical bases (nucleotides). Some CNVs appear first time in the parent's

germ cell called de novo, while others are inherited [6]. Usually, the cell has two copies of each gene; CNV occurs when a part of a gene is deleted or duplicated [7].

Copy number variations affect transcription in humans [8] and have been related to different diseases such as cancer, autism, and schizophrenia [9–11]. All over the world, the most common risk that impends human health is cancer [12]. Cancer is a class of disease which results in irregular growth of cells and is one of the leading causes of human death. The mortality rate of humans due to cancer is about 14.6% each year [13]. Phenotypic variation may also be due to CNVs [6, 14]. The data obtained from CNVs can also be used to classify tumors into malignant and benign [15, 16]. A number of research articles agree that somatic CNVs are mostly associated with the progression of various cancers [17–20].

Machine learning practitioners have proposed a lot of techniques to identify one or multiple types of cancer(s) using various types of genomic data, each with different weaknesses and strengths. During the health checkup, the colonoscopy screening is broadly known for the evaluation of colorectal cancer (CRC) risk, but due to its discomfort and complexity, more reliable and comfortable methods were necessary for the CRC screening. A comprehensive study is presented by Ding et al. [21] about machine learning applications in CNV-based cancer prediction.

Dealing with high-dimensional and heterogeneous data remains a key challenge in healthcare [22]. Traditional methods of machine learning firstly need to perform feature extraction and selection to obtain more useful features from the data and then build prediction models on them. The advancement in deep learning technologies provides effective approaches to obtain end-to-end learning models. Deep learning is a fashionable toolbox and has become popular for big data [23, 24] especially in the field of genomics due to its performance in prediction problems. It is used for many processes such as predicting DNA sequence conversation, identifying enhancers and promoters, and detecting genetic variation from DNA sequencing. The advancement and fruitful applications of deep learning in different fields of genomics reveal that it can be used for cancer classification from CNV data [22, 25–27].

Different computational models for the cancer classification based on copy number variation data are available. The most recently developed model achieves an accuracy up to 85%. The copy number variation data are high dimensional in nature and difficult to handle by the classical machine learning techniques. In this study, we implemented deep learning models that successfully used 24,174 genes of CNV levels to classify six types of cancers: breast adenocarcinoma (BRCA), urothelial bladder carcinoma (BLCA), colon and rectal carcinoma (COAD/READ), glioblastoma multiforme (GBM), kidney renal clear cell carcinoma (KIRC), and head and neck squamous cell (HNSC). The highest obtained average training accuracy is 96%, while testing accuracy is 92%. We have proposed three different deep learning architectures, and all of these models have outperformed state-of-the-art techniques in terms of accuracy, ROC, and precision, while two of our networks have outperformed the state-of-the-art models in terms of recall (see Table 1). So, the contribution of this work is not only to improve the performance (accuracy) of the cancer classifier using an end-to-end model but also to find out which architecture among DNN (deep fully connected neural network), CNN, and RNN is suitable for CNV data. According to our finding, DNN performs better than the rest of the two.

We have discussed the literature review in Section 2, while Section 3 covers the explanation of the dataset and architectures of our models. Section 4 deals with the training process of our models along with obtained results and our findings. Finally, we have concluded our work in Section 5.

## 2. Related Work

Xu et al. [28] have identified the chromosomal alterations in plasma for early detection of CRC. They analyzed the CNVs in cfDNA (cell-free DNA) by using the regular  $z$  score, and the SVM classifier was trained for identification of colon and rectal cancers. The patients with early two stages (I and II) were detected. Brody et al. [29] used blood samples of 8,821 different patients. For feature extraction, they have extracted germline DNA copy number variation data by a single laboratory with an SNP 6.0 array. The gradient boosting algorithm is used to predict breast, ovarian, brain, and colon cancers. Ricatto et al. [30] used a discretizer for feature extraction and a fuzzy rule-based predictor for tumor classification.

In women, breast cancer is the most common type of cancer, which has further subtypes [31]. Pan et al. [32] carried out feature extraction and selection using MCFS (Monte Carlo feature selection). IFS (incremental feature selection) is used to better represent the core CNVs in different subtypes of breast cancer, and then, the dag-stacking model is integrated to detect multiple types of breast cancer. Islam et al. [33] focused on the prediction of molecular subtypes of breast cancer. They performed the experiments to identify binary classes, i.e., estrogen receptor (ER+ and ER-) and multiple classes, i.e., PAM50 (luminal A, luminal B, Her2 enriched, and basal-like). Afterwards, they performed the chi-square test to select the topmost significant genes. For classification, DCNN (deep convolution neural network) was used. Lu et al. [34] also focused on the classification of breast cancer. The authors have introduced a module-based network integrated with genomic data to identify important driver genes in BRCA subtypes. CNV analysis was performed by Li et al. [35] on tumor development. The use case was breast cancer, where they collected data from the TCGA-BRCA project. They searched OMIM (Online Mendelian Inheritance in Man) for most relevant CNVs. They have chosen six candidate genes: ErbB2, AKT2, KRAS, PIK3CA, PTEN, and CCND1. Furthermore, they have constructed two types of distance-based oncogenetic trees to find which of the above candidate genes play a significant role in the development of breast cancer. Their findings showed that ErbB2 has early alteration, while AKT2, KRAS, PIK3CA, PTEN, and CCND1 have late alterations in human breast cancer. Alshibli et al. [36] have proposed deep convolution-based neural networks for CNV data to classify six types of cancer. They have lent the famous computer vision architectures, i.e., ResNet16 and VGG16. Their average accuracy is 86%. They reported that their proposed model has the lowest performance for UCEC (uterine corpus endometrial carcinoma).

To understand the association of CNVs with various types of human cancer, Zhang et al. [37] collected CNV data of different cancer classes consisting of 24,174 genes as features. The feature selection was carried out using minimal

TABLE 1: The average performances of different models along with the state of the art.

S. no	Models	Train Acc	Val Acc (%)	ROC area	Precision	Recall
1	DNN <sub>3</sub>	95%	91	0.99	0.88	0.87
2	DNN <sub>5</sub>	96%	<b>92</b>	0.99	0.89	0.88
3	LSTM	95%	91	0.98	0.89	0.85
4	1D-CNN	88%	90	0.98	0.88	0.85
5	Sana Fekry et al. [38]	—	85.9	0.965	0.852	0.862

redundancy maximal relevance (mRmR) and incremental feature selection (IFS), which resulted in the selection of 200 genes. The dagging model is used for the classification phase of multiple types of cancer. Fekry et al. [38] also worked on these CNV levels of 24,174 genes to classify a set of human cancer types named as breast adenocarcinoma (BRCA), urothelial carcinoma (BLCA), colon and rectal carcinoma (COAD/READ), glioblastoma multiforme (GBM), kidney renal clear cell carcinoma (KIRC), and head and neck squamous cell (HNSC). They selected 16,381 important genes of CNV levels using the filter method (i.e., information gain). For classification, they used seven different classifiers: support vector machine, j48, neural network, random forest, logistic regression, dagging, and bagging. The authors in [39] have contributed to cancer classification using the self-normalizing neural network. They have used Monte Carlo feature selection and incremental feature selection (IFS). They have worked on multiple cancer types and obtained 79% accuracy.

Most recently, researchers are using CNV data along with other modalities such as clinical and/or gene expression data to improve the performance metrics of their models. A contribution is made by researchers in [40] using multimodality data to classify subtypes of breast cancer with the help of the SVM (support vector machine) and RF (random forest). A deep learning model using multi-modality data is used to predict the subtype of breast cancer in [41, 42]. Another deep learning model along with multimodalities of data is used in [43] to predict Alzheimer’s disease. The researchers in [44] have trained their deep learning model on multimodalities to predict therapeutic targets in breast cancer. A comprehensive comparison of multimodalities is presented in [45].

### 3. Materials and Methods

**3.1. Dataset.** For experimentation, we have selected the same dataset used by [38] in order to be compatible in result comparison. The said dataset is composed of six cancer types containing DNA CNVs of 24,174 genes (features/dimensions) for 2916 samples; therefore, the shape of the dataset is  $X_{2916 \times 24174}$  if  $X$  is the input dataset. This dataset was taken from the cBioPortal for Cancer Genomics database [http://cbio.mskcc.org/cancergenomics/pancan\\_tcga/](http://cbio.mskcc.org/cancergenomics/pancan_tcga/). The database contains 11 different types of cancer, and each cancer type has its own samples. The CNV levels were regularized into five distinct values in the database with  $-2$  for homozygous deletion,  $-1$  for heterozygous deletion,  $0$  for diploid,  $1$  for low-level gain, and  $2$  for high-level gain. In this research, we used six different types of cancer, which are listed in Table 2, with names and the number of samples in each class (cancer type).

TABLE 2: The distribution of samples with respect to each cancer type in our dataset.

Sr.	Cancer type	No of samples
0	BRCA (breast carcinoma)	847
1	BLCA (bladder urothelial)	135
2	COAD/READ (colon and rectal adenocarcinoma)	575
3	GBM (glioblastoma multiforme)	563
4	KIRC (kidney renal cell carcinoma)	306
5	HNSC (head and neck squamous cell)	490
Total		2916

### 3.2. Our Proposed Models

**3.2.1. DNN (Deep Fully Connected Neural Network).** An artificial neural network (ANN) is a powerful computational tool that mimics the human brain working behavior [46]. A neural network (NN) consists of a set of neurons arranged in layers such as the input, hidden, and output layer. A single neuron takes an input vector, calculates the weighted sum, and applies the activation function to decide whether it should fire or not. In the fully connected neural network, every neuron of the previous layer is connected to all neurons of the next layer.

For a network of  $L$  number of layers, the  $l^{\text{th}}$  layer is specified by the associated weight matrix  $W^{[l]} \in \mathfrak{R}^{n^{[l+1]} \times n^{[l]}}$ , where  $n^{[l-1]}$  and  $n^{[l]}$  represent the number of neurons in previous and current layers, respectively. The weighted summation of the  $l^{\text{th}}$  layer is given by

$$Z^{[l]} = W^T A^{[l-1]} + b^{[l]}, \quad (1)$$

where  $b \in \mathfrak{R}^{n^{[l]} \times 1}$  is the bias vector and  $A^{[l-1]} \in \mathfrak{R}^{[l-1] \times 1}$  is the activation map of the previous layer.

To speed up the network convergence [47], we have used the batch normalization that scales the  $Z^{[l]}$  in a specified range. Algorithm 1 explains the batch normalization in detail.

In Algorithm 1, the parameters  $\gamma$  and  $\beta$  maintain the expressive power of the network, while  $\epsilon$  is a small positive constant added for computational stability [48]. During the forward pass, an activation map  $A^{[l]}$  is estimated for each layer,  $l = 1, 2, \dots, L$ , to know which neuron should be fired:

$$A^{[l]} = g\left(\tilde{Z}^{[l]}\right), \quad (2)$$

where  $g$  is the activation function. Here, we have used the rectified linear unit (ReLU) as an activation function for all hidden layers:

**Input:**  $Z^{[l]}, \beta, \gamma$   
 $\mu_Z^{[l]} = 1/m \sum_{i=1}^m z_i^{[l]}$  //computing mean of  $Z^{[l]}$   
 $\sigma_Z^{[l]} = \sqrt{\varepsilon + (1/m) \sum_{i=1}^m (z_i^{[l]} - \mu^{[l]})^2}$  //computing standard deviation of  $Z^{[l]}$   
 $\tilde{Z}^{[l]} = Z^{[l]} - \mu_Z^{[l]}/\sigma_Z^{[l]}$   
 $\tilde{Z}^{[l]} = \gamma \tilde{Z}^{[l]} + \beta$  //scaling and shifting  $\tilde{Z}^{[l]}$   
**Return** ( $\tilde{Z}^{[l]}$ )

ALGORITHM 1: Batch normalization.

$$A^{[l]} = \text{ReLU}(\tilde{Z}^{[l]}) = \max(0, \tilde{Z}^{[l]}). \quad (3)$$

The ReLU expedites the training and avoids the vanishing gradient [49]. The last layer in the network is called the output layer (classification layer), which gives the probability of occurrence of different classes. Let there are  $K$  classes, and then, the probability of the dominant class is given by the softmax function:

$$\hat{y} = \arg \max_k \left\{ \frac{e^{z_k^{[L]}}}{\sum_{k=1}^K e^{z_k^{[L]}}} \right\}, \quad (4)$$

where  $z_k^{[L]}$  is the weighted sum of the  $k^{\text{th}}$  unit of output layer  $L$ . In our case, the data contain six classes; thus, we set  $K = 6$ .

In the deep fully connected neural network (DNN) category, we have implemented the networks from shallow to deep by increasing hidden layers one by one. Furthermore, the number of neurons is reduced with a factor of  $\sim 2$  from beginning to end, to achieve dimensionality reduction. We started with a network of three hidden layers as shown in Figure 1 and continued up to seven layers. Aforementioned, we have used ReLU as an activation function in hidden layers with batch normalization and softmax at the output layer. To overcome the issue of overfitting, we have used dropout layers as well. For more details about the dropout layer, read the work of Srivastava et al. [50]. Note that, each input vector  $X$  contain 24,174 features, while the activation map,  $A^{[L-1]}$ , of the last hidden layer contains 150 features, which shows dimensionality reduction. For training, the Adam optimization algorithm along with categorical crossentropy as a loss function is used.

**3.2.2. 1D Convolutional Neural Network.** We have also used the 1D  $D$  convolutional neural network (1D – CNN) for cancer classification. Normally, the CNN contains two parts: (1) convolutional layers that are responsible for feature extraction [51, 52] and (2) the fully connected layer that is responsible for classification. Our proposed 1D – CNN contains two convolutional layers followed by one fully connected layer. Every convolution layer is followed by a stack of max pooling, batch normalization, and dropout layers. Figure 2 presents the detailed architecture of the proposed model.

Note that, the first convolutional layer contains 20 filters, each of size 5, and the ReLU as an activation function. Similarly, the second convolutional layer consists of a stack

of 10 filters, each of size 5, and the ReLU as an activation function. For the activation function in the output layer, we have used softmax (See equation (4)).

**3.2.3. LSTM (Long Short-Term Memory).** LSTM is one of the popular flavors of the RNN (recurrent neural network) with three special gates, i.e., the input/update, forget, and output gate, as shown in Figure 3. The key gate is the forget gate that is used to keep long-term dependency intake. It is the long-term dependency preservation that makes LSTM suitable for sequential data analysis [53].

In our proposed model, we have used 24 LSTM units, ReLU as an activation function followed by a batch normalization layer and then the output layer.

## 4. Results and Discussion

The dataset was split into training and testing with 80% and 20%, respectively, to examine the performance of our proposed models. The methodology that we have adopted is shown in Figure 4. The testing and validation dataset are the same; that is why, validation and testing metrics are the same. The representation learning implicitly exists in the model (s). The worth of representation learning using deep learning has been proved in the literature. As mentioned in Section 3.2, we have implemented three different neural network architectures, to explore their strengths and weaknesses. We have started from the shallow neural network to the deep NN (deep fully connected NN), to LSTM to the 1D-CNN.

We have trained our models up to 200 epochs and plotted the results to check the training status, that is, to find whether the model is underfitted, overfitted, or properly trained.

The obtained training vs. validation accuracies of each model are shown in Figure 5. Given the results in Figure 5 our shallow NN (DNN<sub>3</sub>) and 1D-CNN require more epochs for training, while the remaining deep architectures require less epochs to reach the point where the model starts overfitting. The sign of overfitting is that when the training accuracy improves, while the validation accuracy starts to decline or remains the same. The possible reason behind this behavior is that the deep architecture normally extracts complex but well representative features.

A classwise ROC is shown in Figure 6. The highest ROC, i.e., 1.0 is achieved by all networks for the COAD/READ class, while the average maximum ROC is 0.99 achieved by NN<sub>3</sub> (deep fully connected neural network with 3 layers) and DNN<sub>5</sub> (NN with 5 layers) as shown in Table 1.

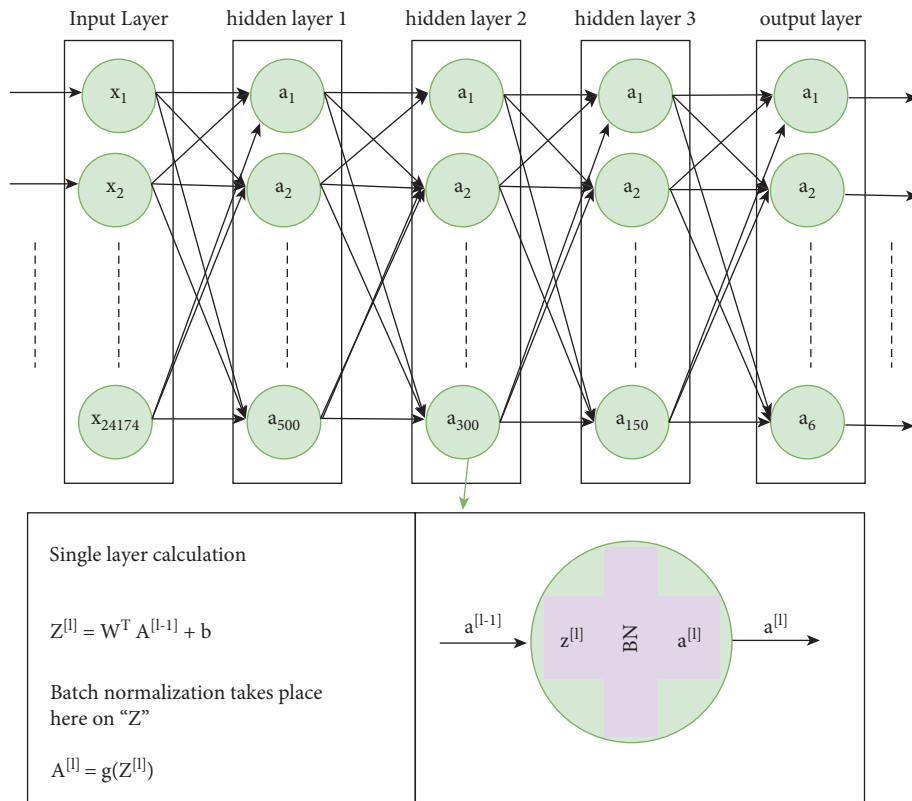


FIGURE 1: The architecture of the fully connected model with three hidden layers.

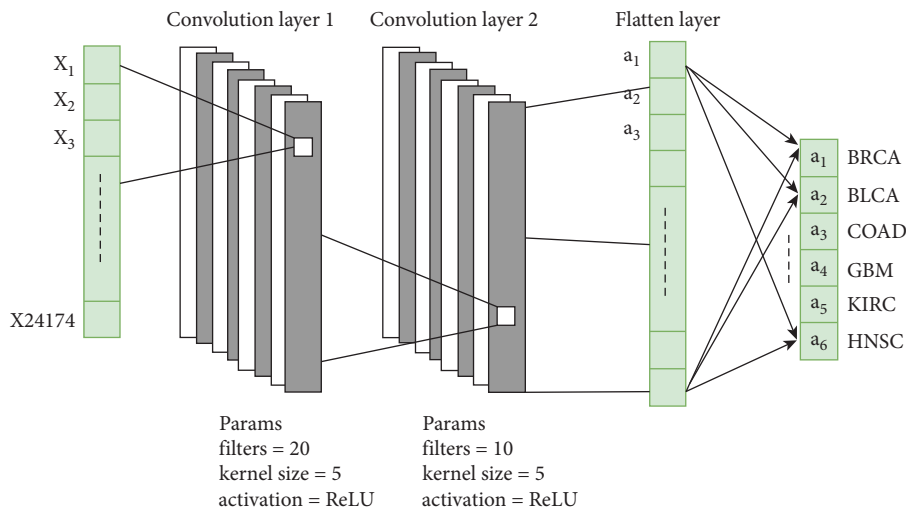


FIGURE 2: 1D convolution-based architecture.

In order to test the performance of our networks for each class (cancer type), we have presented the computed results in Table 3. According to the obtained results, the GBM class is the most complex (difficult) one for our networks, while COAD is the easiest one. The same results can be verified from the confusion matrices given in Tables 4–5.

The average performance measures (in terms of accuracy, precision, recall, and ROC) of all networks are shown in the first four rows of Table 1. The obtained results show

that our DNN architecture has outperformed the rest of our models.

We have compared our computed results with the state-of-the-art models. As mentioned in Table 1, our all networks have outperformed all of our competitors in most of the performance metrics. We have reported only the best results of Sana et al. [38]. Their maximum accuracy is 85% with an ROC area of 0.96, whereas our proposed models achieved the accuracy over 92% with an ROC of 0.99.

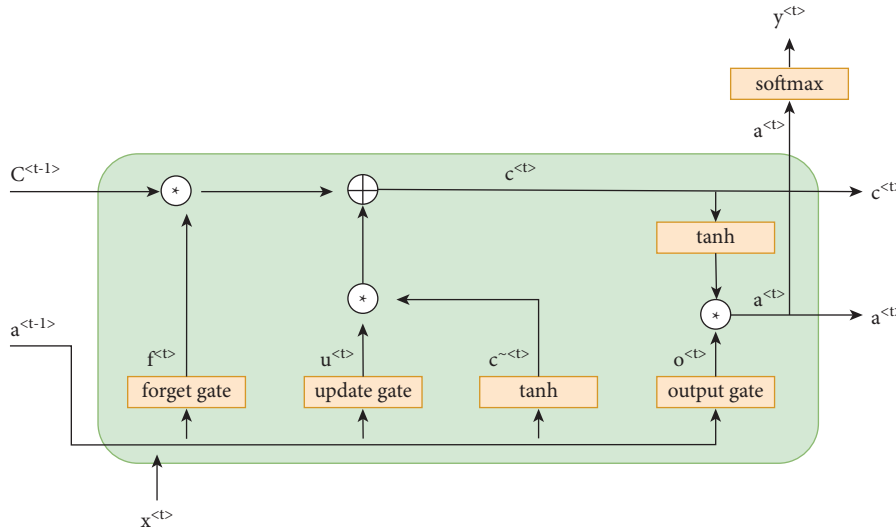


FIGURE 3: LSTM architecture.

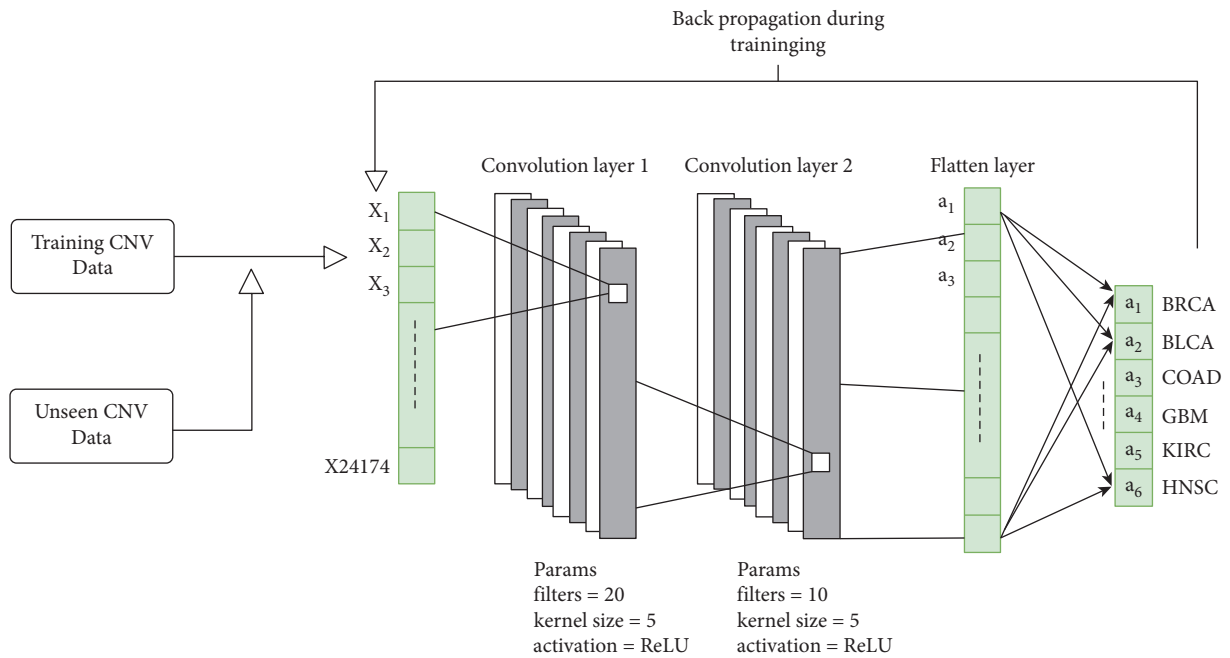


FIGURE 4: Our methodology.

Since Zhang et al. [37] have worked similarly, but their research deals with some different types of cancers, e.g., UCEC (uterine corpus endometrial carcinoma); therefore, the comparison is not compatible, but they have achieved 75.1% accuracy.

In the light of the analysis made on the obtained results, we conclude that due to the small size of the current dataset, very deep neural networks are not beneficial to use as most of our models are converged with the small number of hidden layers. Moreover, the fully

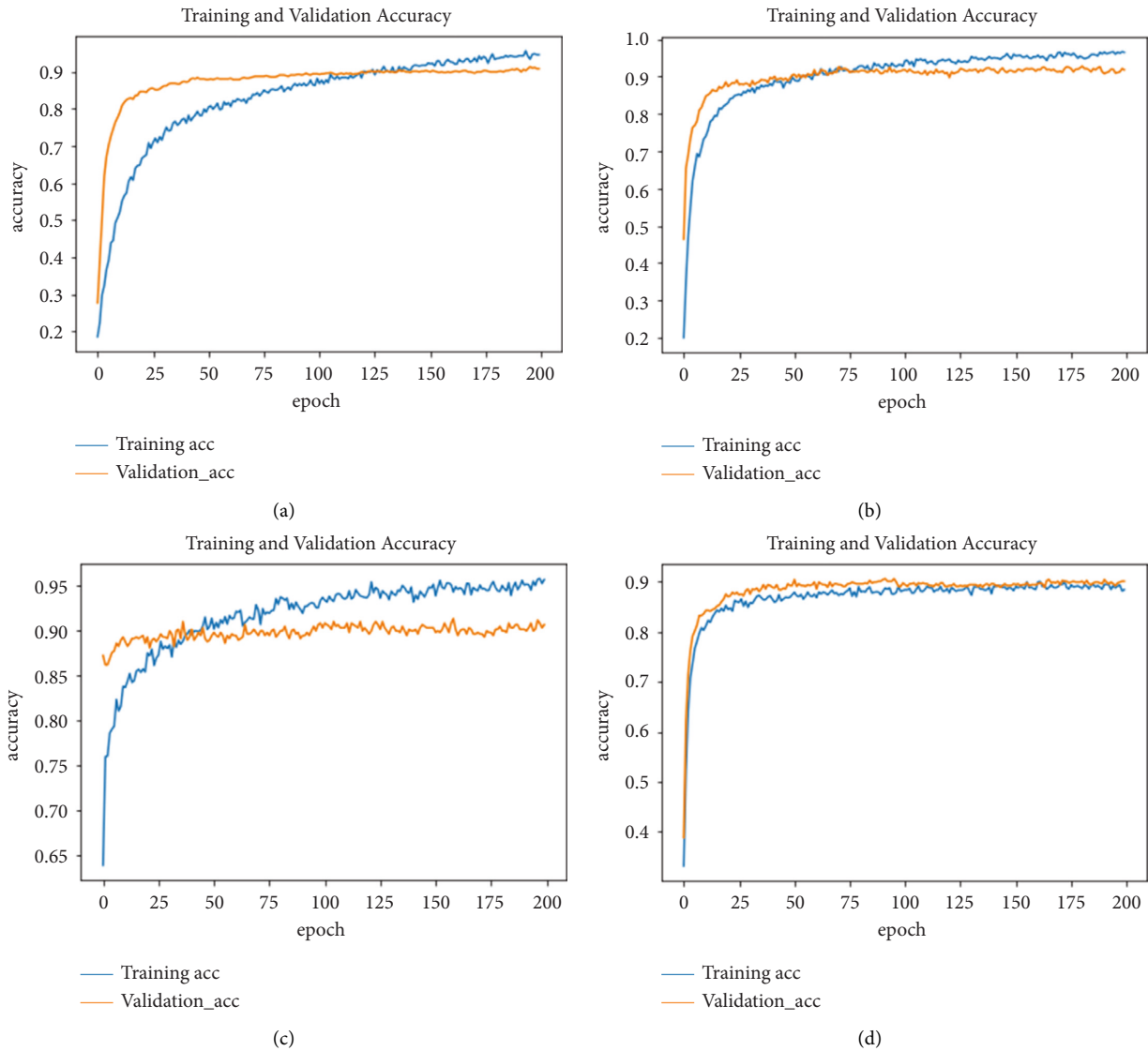


FIGURE 5: Classification of accuracy of different models: (a) DNN<sub>3</sub>, (b) DNN<sub>5</sub>, (c) LSTM, and (d) 1D-CNN.

connected neural network performed better than other flavors such as CNN and RNN for copy number variation (CNV) data (see Table 1). We also found that adding additional layers to a fully connected neural network

(DNN) has a small impact on results. Our obtained results also verify that end-to-end deep learning models are better in representation learning than handcrafted feature extraction (see Table 1)



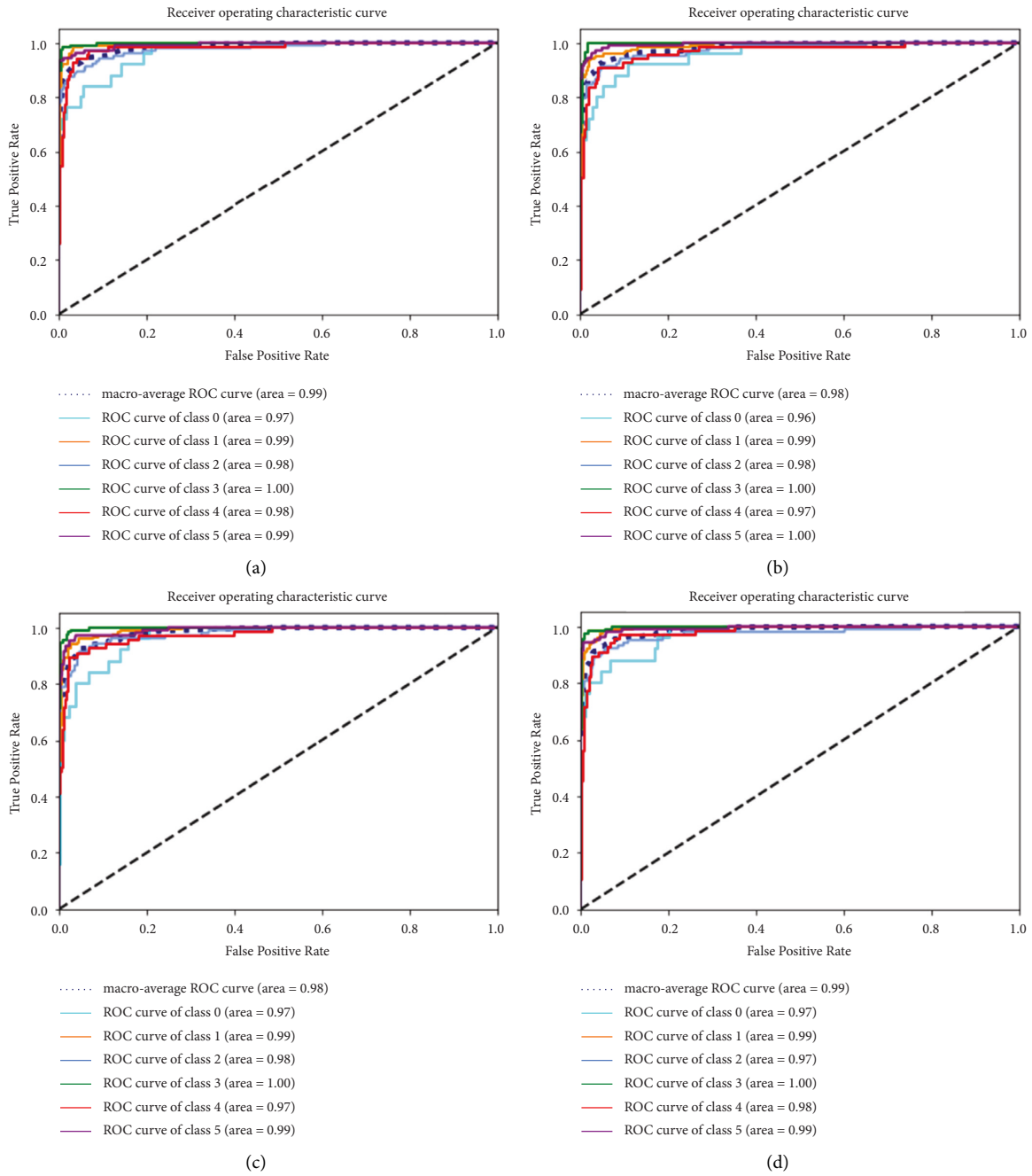


FIGURE 6: ROC of different models on various cancer types: (a) DNN<sub>5</sub>, (b) LSTM, and (c) 1D-CNN.

TABLE 3: The classwise performances of all networks.

Models	GBM (3)	KIRC(4)	HNSC(5)	COAD/READ(2)	BLCA(1)	BRCA(0)
NN						
TP rate	0.68	0.96	0.82	0.98	0.83	0.93
ROC area	0.97	0.99	0.97	1.00	0.98	0.99
Precision	0.77	0.90	0.92	0.93	0.81	0.97
F-measure	0.72	0.93	0.87	0.96	0.82	0.95
Recall	0.68	0.96	0.82	0.98	0.83	0.93
FP rate	0.00	0.01	0.04	0.01	0.02	0.00
DNN						

TABLE 3: Continued.

Models	GBM (3)	KIRC(4)	HNSC(5)	COAD/READ(2)	BLCA(1)	BRCA(0)
TP rate	0.72	0.96	0.85	0.98	0.85	0.94
ROC area	0.97	0.98	0.99	1.00	0.98	0.99
Precision	0.75	0.93	0.94	0.94	0.85	0.93
<i>F</i> -measure	0.73	0.94	0.89	0.96	0.85	0.94
Recall	0.72	0.96	0.85	0.98	0.85	0.94
FP rate	0.01	0.02	0.01	0.01	0.01	0.01
LSTM						
TP rate	0.52	0.95	0.85	0.98	0.88	0.92
ROC area	0.96	0.99	0.98	1.00	0.97	1.00
Precision	0.87	0.91	0.93	0.92	0.79	0.95
<i>F</i> -measure	0.65	0.93	0.88	0.95	0.83	0.94
Recall	0.68	0.94	0.84	0.96	0.79	0.91
FP rate	0.52	0.95	0.85	0.98	0.88	0.92
1D-CNN						
TP rate	0.64	0.93	0.92	0.96	0.77	0.91
ROC area	0.97	0.99	0.97	1.00	0.97	0.99
Precision	0.84	0.93	0.81	0.93	0.86	0.94
<i>F</i> -measure	0.73	0.93	0.86	0.94	0.82	0.92
Recall	0.64	0.93	0.92	0.96	0.77	0.91
FP rate	0.00	0.02	0.04	0.01	0.01	0.01

TABLE 4: Confusion matrix for training data.

	BRCA(0)	BLCA(1)	COAD/READ(2)	GBM (3)	KIRC(4)	HNSC(5)
BRCA(0)	<b>109</b>	0	1	0	0	0
BLCA(1)	0	<b>673</b>	6	0	0	0
COAD/READ(2)	0	1	<b>470</b>	0	0	0
GBM (3)	0	0	2	<b>446</b>	0	0
KIRC(4)	0	0	7	0	<b>233</b>	0
HNSC(5)	0	0	3	0	0	<b>381</b>

TABLE 5: The confusion matrix for testing data.

	BRCA(0)	BLCA(1)	COAD/READ(2)	GBM (3)	KIRC(4)	HNSC(5)
BRCA(0)	<b>15</b>	1	2	2	3	2
BLCA(1)	0	<b>158</b>	3	3	2	2
COAD/READ(2)	0	3	<b>94</b>	1	4	2
GBM (3)	0	0	1	<b>113</b>	1	0
KIRC(4)	1	5	1	0	<b>55</b>	4
HNSC(5)	1	0	3	1	1	<b>100</b>

## 5. Conclusion and Future Directions

Copy number variations are related to different human diseases, such as cancer, autism, and schizophrenia. In this paper, we classified six different types of cancers by using copy number variation data. We have proposed three different neural network architectures to make the classification process end-to-end. Moreover, we have effectively used the data-hungry nature of the deep neural network and we have not used the feature engineering (handcrafted feature extraction) step as used by most of the researchers to save computational time. Our achieved testing accuracies are 91%, 92%, 90%, and 91% by using CNV levels of 24,174 genes. Our work testifies that the CNVs of these genes play a crucial role in classifying human cancers. In the future, we aim to work on the other types of cancer as well.

## Data Availability

The data are publicly available at this link: [http://cbio.mskcc.org/cancergenomics/pancan\\_tcga](http://cbio.mskcc.org/cancergenomics/pancan_tcga).

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was supported by EIAS (Emerging Intelligent Autonomous Systems) Data Science Lab, Prince Sultan University, KSA. The authors would like to thank the EIAS Data Science Lab and Prince Sultan University for their encouragement, support, and the facilitation of resources needed and funding to complete this work.

## References

- [1] A. Thapar and M. Cooper, "Copy number variation: what is it and what has it told us about child psychiatric disorders?" *Journal of the American Academy of Child & Adolescent Psychiatry*, vol. 52, no. 8, pp. 772–774, 2013.
- [2] N. M. Williams, I. Zaharieva, A. Martin et al., "Rare chromosomal deletions and duplications in attention-deficit hyperactivity disorder: a genome-wide analysis," *The Lancet*, vol. 376, no. 9750, pp. 1401–1408, 2010.
- [3] T. Y. Leung, R. K. Pooh, C. C. Wang, T. K. Lau, and K. W. Choy, "Classification of pathogenic or benign status of CNVS detected by microarray analysis," *Expert Review of Molecular Diagnostics*, vol. 10, no. 6, pp. 717–721, 2010.
- [4] Z. Zhang, H. Cheng, X. Hong et al., "Ensemblecnv: an ensemble machine learning algorithm to identify and genotype copy number variation using snp array data," *Nucleic Acids Research*, vol. 47, no. 7, p. e39, 2019.
- [5] J. Zhang, L. Feuk, G. Duggan, R. Khaja, and S. Scherer, "Development of bioinformatics resources for display and analysis of copy number and other structural variants in the human genome," *Cytogenetic and Genome Research*, vol. 115, no. 3–4, pp. 205–214, 2006.
- [6] I. Ostrovnyaya, G. Nanjangud, and A. B. Olshen, "A classification model for distinguishing copy number variants from cancer-related alterations," *BMC Bioinformatics*, vol. 11, no. 1, p. 297, 2010.
- [7] P. Stankiewicz and J. R. Lupski, "Structural variation in the human genome and its role in disease," *Annual Review of Medicine*, vol. 61, no. 1, pp. 437–455, 2010.
- [8] C. Chiang, A. J. Scott, J. R. Davis et al., "The impact of structural variation on human gene expression," *Nature Genetics*, vol. 49, no. 5, pp. 692–699, 2017.
- [9] G. A. Erikson, N. Deshpande, B. G. Kesavan, and A. Torkamani, "Sg-adviser cnv: copy-number variant annotation and interpretation," *Genetics in Medicine*, vol. 17, no. 9, pp. 714–718, 2015.
- [10] O. Pös, J. Radvanszky, G. Buglyó et al., "Dna copy number variation: main characteristics, evolutionary significance, and pathological aspects," *Biomedical Journal*, vol. 44, no. 5, pp. 548–559, 2021.
- [11] E. Sarkar, E. Chielle, G. Gursoy, L. Chen, M. Gerstein, and M. Maniatakos, "Scalable privacy-preserving cancer type prediction with homomorphic encryption," 2022, <http://arXiv.org/abs/2204.05496>.
- [12] Y. Sun, S. Zhu, K. Ma et al., "Identification of 12 cancer types through genome deep learning," *Scientific Reports*, vol. 9, no. 1, Article ID 17256, 2019.
- [13] Y. Yuan, Y. Shi, X. Su et al., "Cancer type prediction based on copy number aberration and chromatin 3d structure with convolutional neural networks," *BMC Genomics*, vol. 19, no. S6, p. 565, 2018.
- [14] C. A. Brownstein, R. S. Smith, L. H. Rodan et al., "Rc1l copy number variants are associated with a range of neuropsychiatric phenotypes," *Molecular Psychiatry*, vol. 26, no. 5, pp. 1706–1718, 2021.
- [15] A. Mahas, K. Potluri, M. N. Kent, S. Naik, and M. Markey, "Copy number variation in archival melanoma biopsies versus benign melanocytic lesions," *Cancer Biomarkers*, vol. 16, no. 4, pp. 575–597, 2016.
- [16] C. F. Ebbelaar, A. M. R. Schrader, M. van Dijk et al., "Towards diagnostic criteria for malignant deep penetrating melanocytic tumors using single nucleotide polymorphism array and next-generation sequencing," *Modern Pathology*, vol. 2021, pp. 1–11, 2022.
- [17] L. Yang, Y. Z. Wang, H. H. Zhu et al., "Prame gene copy number variation is related to its expression in multiple myeloma," *DNA and Cell Biology*, vol. 36, no. 12, pp. 1099–1107, 2017.
- [18] Y. S. Huang, W. B. Liu, F. Han et al., "Copy number variations and expression of mpdz are prognostic biomarkers for clear cell renal cell carcinoma," *Oncotarget*, vol. 8, no. 45, pp. 78713–78725, 2017.
- [19] C. Zhou, W. Zhang, W. Chen et al., "Integrated analysis of copy number variations and gene expression profiling in hepatocellular carcinoma," *Scientific Reports*, vol. 7, no. 1, Article ID 10570, 2017.
- [20] J. Samulin, Y. J. Arnoldussen, Y. Erdem et al., "Copy number variation, increased gene expression, and molecular mechanisms of neurofascin in lung cancer," *Molecular Carcinogenesis*, vol. 56, no. 9, pp. 2076–2085, 2017.
- [21] X. Ding, S. Y. Tsang, S. K. Ng, and H. Xue, "Application of machine learning to development of copy number variation-based prediction of cancer risk," *Genomics Insights*, vol. 7, pp. GEI.S15002–11, 2014.
- [22] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, "Deep learning for healthcare: review, opportunities and challenges," *Briefings in Bioinformatics*, vol. 19, no. 6, pp. 1236–1246, 2018.
- [23] B. Jan, H. Farman, M. Khan et al., "Deep learning in big data analytics: a comparative study," *Computers & Electrical Engineering*, vol. 75, pp. 275–287, 2019.
- [24] M. Khan, B. Jan, and H. Farman, *Deep Learning: Convergence to Big Data Analytics*, Springer, Berlin, Germany, 2019.
- [25] C. Angermueller, T. Pärnamaa, L. Parts, and O. Stegle, "Deep learning for computational biology," *Molecular Systems Biology*, vol. 12, no. 7, p. 878, 2016.
- [26] Y. Hu, L. Zhao, Z. Li, X. Dong, T. Xu, and Y. Zhao, "Classifying the multi-omics data of gastric cancer using a deep feature selection method," *Expert Systems with Applications*, vol. 200, Article ID 116813, 2022.
- [27] D. Khan and S. Shedole, "Leveraging deep learning techniques and integrated omics data for tailored treatment of breast cancer," *Journal of Personalized Medicine*, vol. 12, no. 5, p. 674, 2022.
- [28] J.-F. Xu, Q. Kang, X.-Y. Ma et al., "A novel method to detect early colorectal cancer based on chromosome copy number variation in plasma," *Cellular Physiology and Biochemistry*, vol. 45, no. 4, pp. 1444–1454, 2018.
- [29] C. Toh and J. P. Brody, "Analysis of copy number variation from germline dna can predict individual cancer risk," *bioRxiv*, Article ID 303339, 2018.
- [30] M. Ricatto, M. Barsacchi, and A. Bechini, "Interpretable cnv-based tumour classification using fuzzy rule based classifiers," in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, pp. 54–59, NY, USA, April 2018.
- [31] A. Szymiczek, A. Lone, and M. R. Akbari, "Molecular intrinsic versus clinical subtyping in breast cancer: a comprehensive review," *Clinical Genetics*, vol. 99, no. 5, pp. 613–637, 2021.
- [32] X. Pan, X. Hu, Y.-H. Zhang et al., "Identification of the copy number variant biomarkers for breast cancer subtypes," *Molecular Genetics and Genomics*, vol. 294, no. 1, pp. 95–110, 2019.
- [33] M. M. Islam, R. Ajwad, C. Chi, M. Domaratzki, Y. Wang, and P. Hu, "Somatic copy number alteration-based prediction of molecular subtypes of breast cancer using deep learning model," *Canadian Conference on Artificial Intelligence*, vol. 10233, pp. 57–63, 2017.

- [34] X. Lu, X. Li, P. Liu, X. Qian, Q. Miao, and S. Peng, "The integrative method based on the module-network for identifying driver genes in cancer subtypes," *Molecules*, vol. 23, no. 2, p. 183, 2018.
- [35] X.-C. Li, C. Liu, T. Huang, and Y. Zhong, "The occurrence of genetic alterations during the progression of breast carcinoma," *BioMed Research International*, vol. 2016, Article ID 5237827, 5 pages, 2016.
- [36] A. AlShibli and H. Mathkour, "A shallow convolutional learning network for classification of cancers based on copy number variations," *Sensors*, vol. 19, no. 19, p. 4207, 2019.
- [37] N. Zhang, M. Wang, P. Zhang, and T. Huang, "Classification of cancers based on copy number variation landscapes," *Biochimica et Biophysica Acta (BBA) - General Subjects*, vol. 1860, no. 11, pp. 2750–2755, 2016.
- [38] S. F. A. Elsadek, M. A. A. Makhlof, and M. A. Aldeen, "Supervised classification of cancers based on copy number variation," *Advances in Intelligent Systems and Computing*, vol. 845, pp. 198–207, 2018.
- [39] J. Li, Q. Xu, M. Wu, T. Huang, and Y. Wang, "Pan-cancer classification based on self-normalizing neural networks and feature selection," *Frontiers in Bioengineering and Biotechnology*, vol. 8, p. 766, 2020.
- [40] A. El-Nabawy and N. A. Belal, "A feature-fusion framework of clinical, genomics, and histopathological data for metabric breast cancer subtype classification," *Applied Soft Computing*, vol. 91, Article ID 106238, 2020.
- [41] Y. Lin, W. Zhang, H. Cao, G. Li, and W. Du, "Classifying breast cancer subtypes using deep neural networks based on multi-omics data," *Genes*, vol. 11, no. 8, p. 888, 2020.
- [42] T. Liu, J. Huang, T. Liao, R. Pu, S. Liu, and Y. Peng, "A hybrid deep learning model for predicting molecular subtypes of human breast cancer using multimodal data," *IRBM*, vol. 43, no. 1, pp. 62–74, 2022.
- [43] S. Dwivedi, T. Goel, M. Tanveer, R. Murugan, and R. Sharma, "Multi-modal fusion based deep learning network for effective diagnosis of Alzheimer's disease," *IEEE MultiMedia*, p. 1, 2022.
- [44] X. Pan, B. Burgman, N. Sahni, and S. Yi, "Deep learning based on multi-omics integration identifies potential therapeutic targets in breast cancer," *bioRxiv*, pp. 2–23, 2022.
- [45] F. Carrillo-Perez, J. C. Morales, D. Castillo-Secilla, A. Guillen, I. Rojas, and L. J. Herrera, "Comparison of fusion methodologies using CNV and RNA-seq for cancer classification: a case study on non-small-cell lung cancer," *Bioengineering and Biomedical Signal and Image Processing*, vol. 12940, pp. 339–349, 2021.
- [46] M. O. Okwu and L. K. Tartibu, "Artificial neural network," *Metaheuristic Optimization: Nature-Inspired Algorithms Swarm and Computational Intelligence, Theory and Applications*, vol. 927, pp. 133–145, 2021.
- [47] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," pp. 448–456, 2015, <https://arxiv.org/abs/1502.03167>.
- [48] I. Goodfellow, Y. Bengio, and A. Courville, "Deep Learning," MIT Press, 2016, <http://www.deeplearningbook.org>.
- [49] Z. Hu, J. Zhang, and Y. Ge, "Handling vanishing gradient problem using artificial derivative," *IEEE Access*, vol. 9, pp. 22371–22377, 2021.
- [50] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [51] F. Li, M. Liu, Y. Zhao et al., "Feature extraction and classification of heart sound using 1d convolutional neural networks," *EURASIP Journal on Applied Signal Processing*, vol. 59, no. 1, pp. 59–11, 2019.
- [52] H. Yang, C. Meng, and C. Wang, "Data-driven feature extraction for analog circuit fault diagnosis using 1-d convolutional neural network," *IEEE Access*, vol. 8, pp. 18305–18315, 2020.
- [53] J. Zhao, F. Huang, J. Lv et al., "Do RNN and LSTM have long memory?" pp. 11365–11375, 2020, <https://arxiv.org/pdf/2006.03860.pdf>.

## Research Article

# An Analysis of New Feature Extraction Methods Based on Machine Learning Methods for Classification Radiological Images

**Firoozeh Abolhasani Zadeh <sup>1</sup>, Mohammadreza Vazifeh Ardalani,<sup>2</sup> Ali Rezaei Salehi,<sup>3</sup> Roza Jalali Farahani,<sup>4</sup> Mandana Hashemi,<sup>5</sup> and Adil Hussein Mohammed<sup>6</sup>**

<sup>1</sup>Department of Surgery, Faculty of Medicine, Kerman University of Medical Sciences, Kerman, Iran

<sup>2</sup>Robotics Research Laboratory, Center of Excellence in Experimental Solid Mechanics and Dynamics, School of Mechanical Engineering, Iran University of Science and Technology, Tehran, Iran

<sup>3</sup>Industrial Engineering Department, Technical and Engineering Faculty, University of Science and Culture, Tehran, Iran

<sup>4</sup>Department of Electrical Engineering, Islamic Azad University, Tehran, Iran

<sup>5</sup>School of Industrial and Information Engineering, Politecnico di Milano University, Milan, Italy

<sup>6</sup>Department of Communication and Computer Engineering, Faculty of Engineering, Cihan University-Erbil, Erbil, Kurdistan Region, Iraq

Correspondence should be addressed to Firoozeh Abolhasani Zadeh; [f.abolhasani@kmu.ac.ir](mailto:f.abolhasani@kmu.ac.ir)

Received 1 January 2022; Revised 2 February 2022; Accepted 8 March 2022; Published 25 May 2022

Academic Editor: Mohammed A. A. Al qaness

Copyright © 2022 Firoozeh Abolhasani Zadeh et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The lungs are COVID-19's most important focus, as it induces inflammatory changes in the lungs that can lead to respiratory insufficiency. Reducing the supply of oxygen to human cells negatively impacts humans, and multiorgan failure with a high mortality rate may, in certain circumstances, occur. Radiological pulmonary evaluation is a vital part of patient therapy for the critically ill patient with COVID-19. The evaluation of radiological imagery is a specialized activity that requires a radiologist. Artificial intelligence to display radiological images is one of the essential topics. Using a deep machine learning technique to identify morphological differences in the lungs of COVID-19-infected patients could yield promising results on digital images of chest X-rays. Minor differences in digital images that are not detectable or apparent to the human eye may be detected using computer vision algorithms. This paper uses machine learning methods to diagnose COVID-19 on chest X-rays, and the findings have been very promising. The dataset includes COVID-19-enhanced X-ray images for disease detection using chest X-ray images. The data were gathered from two publicly accessible datasets. The feature extractions are done using the gray level co-occurrence matrix methods. *K*-nearest neighbor, support vector machine, linear discrimination analysis, naïve Bayes, and convolutional neural network methods are used for the classification of patients. According to the findings, convolutional neural networks' efficiency linked to imaging modalities with fewer human involvements outperforms other traditional machine learning approaches.

## 1. Introduction

COVID-19 revealed flaws in many countries' healthcare services, and the failure of these systems to treat patients has generated concern. The lack of accuracy in clinical detection methods is significant for COVID-19's rapid dissemination [1]. Molecular methodologies, such as computational real reverse transcription-polymerase chain reaction (rRT-PCR) [2], as well as other techniques, such as serologic tests [3] and

swab testing of the throat [4, 5], are employed and widely utilized to diagnose COVID-19. Researchers also used chest radiographs (X-rays) and chest computed tomography (CT) scans to help indicate abnormalities characteristic of various lung diseases, including COVID-19. CT scans and X-ray examinations may be used as a critical screening method to assess COVID-19 seriousness, track infectious patients' emergency cases, and forecast COVID-19 progression [6, 7]. However, time is always short in such situations. These trials

cannot be carried out using the current standard manual diagnosis [8]. Deep learning (DL) is a branch of machine learning (ML) that learns the highest-level abstractions from findings using hierarchical structures. DL is utilized to solve artificial intelligence challenges due to the multiple computational levels. The layers of DL are created from data using a learning process rather than by individual technologists, which is a crucial feature of the DL approach. Its benefits, such as higher achievement, blended function learning with an end-to-end learning scheme, and the potential to monitor composite knowledge challenges, have achieved relevant therapeutic effectiveness in healthcare. DL algorithms are also being employed to track and evaluate coronavirus pandemics, thanks to their extensive application in the research of medical recordings. Some theorists and simulation engineers used autoencoders, convolutional neural networks (CNN), and generative adversarial networks to analyze the coronavirus pandemic. Mathematical and predictive methods may estimate human loss and predict death over a specific time frame or up to the end of the pandemic. Since statistical simulations do not consider all facets of the pandemic, they cannot make more precise predictions. To construct powerful models and resources that support clinicians detect COVID-19 infection, scientists and modeling engineers have widely used intelligent computation methods in healthcare. The government is taking the necessary measures to prohibit the COVID-19 pandemic from spreading and advancing intelligent computing methods in healthcare. As a result, several ML approaches are being used to tackle the latest COVID-19 pandemic successfully. de Moraes Batista et al. [9] proposed that various machine learning methods are used for the prognosis of novel coronaviruses. Moreover, training the network with 70% reference data and 30% test data measured the accuracy of different ML techniques. With 68% sensitivity, 85% specificities, 85% AUC, and 0.16 Brier scores, the support vector machine (SVM) outperforms random forests, logistic regression, and gradient boosting tree techniques. ML approaches have several drawbacks, including the heavy use of patient data [10], inconsistency, dependence on temporal data, paucity, discrepancy [11], and the failure to produce accurate forecasts due to high dimensionality [12]. The generalized logistic growth model was utilized by Ahmadi et al. [13] to predict the subepidemic waves of the COVID-19 outbreak. Two, three, and four-wave phenomena were predicted using the formula. Centered on a lung X-ray map, Hassantabar et al. used an ML algorithm to detect infected COVID-19 patient tissue [14, 15]. This discovery can also be utilized to observe and manage patients' progress in contaminated areas [16–20].

This paper employs machine learning methods to classify COVID-19 X-ray images. The feature extraction process is the first and most crucial step in the classification process in machine learning. Even though many deep learning algorithms employ input photos to train the network, the GLCM feature extraction approach shown in this study is successful. As opposed to featuring extraction techniques with an image input, the essential point in feature extraction methods with picture input is to lower the processing time. The size of the

output network is significantly less than with image input. The last phase involves the application of machine learning technologies and the extraction of network performance data. The plans are compared in the confusion matrix and ROC curve. For the classification of patients, KNN, SVM, LDA, NB, and CNN methods are used.

## 2. Literature Review

By learning from basic depictions, deep learning strategies can clarify complicated problems. DL methods have become popular because they learn exact representations and the property of studying information in a fundamental method where several layers are used sequentially [21]. DL techniques are usually employed in medical science, biomedical science [20, 22], innovative health [23, 24], drug delivery [25], and medical image recognition [26, 27], among others. It is also commonly used to make an automated COVID-19 diagnosis. Data analysis, data processing, feature extraction and classification, and performance assessment are all stages in deep learning-based systems [21, 28]. A pretrained method has already been trained in fields related to the application's context. Weight and prejudice were retrained in transfer learning from a limited training network to a new study network. In particular, training big data takes a long time and needs much computing capital [21]. Xu et al. evaluated a novel approach employing multitask joint training algorithm for sectioning and classifying tongue images using a deep convolutional neural network [29].

In another research by using support vector collection data, a multiple kernel-based fuzzy SVM model was developed to predict DNA-binding proteins by Zou et al. [30]. Using a pretrained transfer learning model, the facility speeds up the convergence with network generalization [31]. In transfer learning, multiple pretraining networks are required for the huge CNN. Some of the pretrained network of COVID-19 classifications include AlexNet [32], GoogleNet [33], SqueezeNet [34], various variants of visual geometry group (VGG) [35], various forms of ResNet [36], Xception [37], various forms of inception [38], various types of MobileNet [39], DenseNet [40], U-Net [41, 42], and others. Using transfer learning, detecting 0 in CT and X-ray images has been effectively extended. 3D CT images are handled distinctly other than colored X-ray. 3D CT images have a set number of slices based on the computer and configuration (16, 32, 59, 128, etc.). Individual slices of nature may be greyscale or color photographs. Usually, the slices are isolated and treated as individual pictures [43]. The slices with the most lung areas are included, while the remaining slices are removed. Rezaei et al. created a lithological cartography in the Sangan region of Northeastern Iran using satellite data and image processing technologies [44]. During COVID-19, Arenliu et al. have statistically analyzed the building of online and telephone psychological first aid services in a low-resource setting [45]. Rezaei et al. introduced a data-driven approach for segmenting hand parts on depth maps that did not need any additional effort to acquire segmentation labels [46]. In [47], case study features obtained from the slices are utilized to optimize the pretrained network. Also, U-Net has

been utilized to segment and reduce features from various regions of interest (ROI) in 3D CT in some cases [48]. Rehman and Lela investigated how to manage crises during the COVID-19 pandemic [50]. Radulescu and Cavanagh adapted their SEIR standard model [51] to investigate the complicated hierarchical compartments and epidemiologic factors of COVID-19. They reviewed the new management policies of the pandemic, mutual isolation, travel prohibition, interruptions, and closings, to produce predictions and assess the feasibility of containment measures. The aim of this study was to estimate the distribution gap in COVID-19 by means of a hybrid of SEIR models and regression models applying data from the John Hopkins University on COVID-19 [52]. Sadeghipour et al. conducted different research in which they compared the Expert Clinical System for Diagnosing Obstructive Sleep Apnea to the XCSR classifier [54]. According to the study by Zhang et al., a privacy-preserving algorithm is implemented for querying clinical pathways in e-healthcare systems [55]. Liu et al. published a self-supervised CycleGAN method for super-resolving ultrasound images with perception consistency [56]. Several ways to reduce the negative economic effects of COVID-19 were presented by Mahmoudi et al. [57]. Chaudhary et al. proposed the Fourier-Bessel series expansion-based decomposition technique, a Fourier-Bessel series expansion domain application of the wavelet packet decomposition methodology. Using a transfer learning technique, the subband pictures are utilized for training multiple pretrained CNN models independently. A feature set is obtained by fusing the in-depth features of each channel. Several classifiers distinguish pneumonia caused by COVID-19 from other viral and bacterial pneumonia and healthy patients with the recovered feature vector. The COVID-19 categorization based on chest X-rays requires a radiology specialist and a substantial amount of time, both valuable commodities when COVID-19 illness spreads rapidly. As a result, creating an automated analytic technique is desirable to save time for medical experts. The research discussed in [59] uses the susceptible exposed infected removed (SEIR) method for physical estimation and evaluation distancing. If the process of return to work started in April, physical distance measures were successful. This study investigates the relationship between temperature and humidity in relation to the transmission of COVID-19 [60]. Alyasseri et al. [57] divided the research tracks into two categories: deep learning (DL) and machine learning (ML), and they presented COVID-19 public datasets that had been produced and retrieved from different nations. The measurements used to evaluate diagnostic approaches have been compared, and a thorough explanation has been provided. The study served as a guide for the research community on the planned development of machine learning for COVID-19. It served as an inspiration for their work on subsequent developments.

Kumar et al. [76] investigated various artificial intelligence-based strategies to apply these techniques to the prediction and diagnosis of COVID-19 illness. Their ideas for future study and their facilitation of knowledge collecting and formulation on the application of artificial intelligence

approaches for dealing with the COVID-19 outbreak and its effects were much appreciated. Using computed CT scans, Akram et al. [77] demonstrated an automated approach for the quick diagnosis of COVID-19 infection. They used discrete wavelet transform and extended segmentation-based fractal texture analysis methods to extract the essential characteristics from the fractal texture dataset. Khan et al. [78] presented a strategy for enhancing contrast by combining top-hat and Wiener filters, which they called “contrast enhancement.” Two deep learning models that had already been trained were used and fine-tuned according to the target classes. The features were then retrieved and fused using a parallel fusion approach—the parallel positive correlation—to achieve the best results. The entropy-controlled firefly optimization approach was used to identify the most optimal characteristics. Nejatishahidin et al. [79] proposed a new pose estimation method that can be applied to previously undetected environments. A framework for identifying 15 different forms of chest illnesses, including the COVID-19 disease, was suggested by Rehman et al. [80]. The framework was based on the use of a chest X-ray modality. This architecture enhanced the accuracy of COVID-19 identification while simultaneously increasing the prediction rates for other chest disorders. According to one suggestion, face masks can be used nationwide and applied promptly (see Table 1).

### 3. Methods and Materials

*3.1. Feature Extraction.* The extraction feature decreases the number of tools used to interpret large data sets accurately. The vast number of variables involved is one of the critical problems of sophisticated data analytics. It would help if you had plenty of memory and computing power for working with an extensive range of variables or required an algorithm to classify the exhibits and use them for new samples. Feature extraction is a wide-ranging concept to create dynamic mixes of variables to tackle these problems with absolute accuracy. Texture analysis sought to identify a reliable way to represent the specific properties of textures in a simplified but specific way for correct object detection and division. The surface is essential in image analysis and pattern recognition. Texture extraction is only used in a few processor architectures [75].

A gray surface incidence matrix is developed in this study to achieve statistical texture characteristics. The texture properties of the observed substances are calculated from the statistical light intensity distribution of specific positions in the statistical texture analysis concerning each other. The numbers are classified as first, second, or higher order, depending on the number of pixels for each combination. The gray level matrix (GLCM) is a technique to calculate statistical texture characteristics of the second order. In several recent studies, this method has been used for third and higher-order textures that recognize the relationships between three or more physically feasible pixels but are seldom used because time and perception complexity are computed. GLCM has 22 features, which are listed as follows [81]:

TABLE 1: Literature review related to deep learning method for classification of COVID-19.

Approaches	Dataset	Volume	TPR	Acc
COVID-Net [58]	COVIDx test	13800	0.871	0.926
ResNet50; InceptionV3; Inception-ResNetV2 [59]	GitHub	100	—	98%; 97%; 87%
COVNet [60]	Proprietary datasets	4356	0.87	
Deep learning with X-ray [61]	Proprietary datasets	448	0.986 6	0.967 8
COVIDX-Net (VGG19 and DenseNet201) [62]	Proprietary datasets	50	—	0.9
Barstugan et al. [63]	Proprietary datasets	150	—	0.996 8
ResNet50 and SVM [64]	GitHub, Kaggle, and Open-i	158	97. 29%	0.953 8
SVM and random forests [65]	Hospital Israelita Albert Einstein in São Paulo	—	0.067 7	0.847
MLT and SVM [66]	Montgomery County X-ray Set and COVID Chest X-ray Set and COVID Chest X-ray dataset master	40	0.957 6	0.974 8
Li et al. [67]	Proprietary	—	0.8	0.87
SMOTE [68]	Chest X-ray images (Pneumonia)1 and COVID-19 public dataset from Italy	5840	0.967	0.966
Probabilistic model [69]	Kaggle benchmark dataset	51	—	0.994
NLR&RDW-SD [70]	Jingzhou Central Hospital	—	0.9	0.857
RF-based model [71]	Proprietary	—	—	0.875
SMOTE [68]	Chest X-ray images (Pneumonia)1 and COVID-19 public dataset from Italy	5840	0.932	0.931
iSARF [72]	3 University Hospitals (Tongji, Shanghai, Fudan)	—	0.907	0.879
SMOTE [68]	Chest X-ray images (Pneumonia)1 and COVID-19 public dataset from Italy	5840	0.947	0.947
Modified U-Net structure [73]	SIRM	110	—	0.79
Attention U-Net with an adversarial critic model [74]	JSRT, Montgomery, and Shenzhen	1047	—	0.96
InfNet and the Semi-Inf-Net [75]	CCOVID-19 CT segmentation and COVID-19 CT/X-ray collection	1600	0.725	—

- (1) Energy
- (2) Entropy
- (3) Dissimilarity
- (4) Contrast/inertia
- (5) Inverse difference
- (6) Correlation
- (7) Homogeneity
- (8) Autocorrelation
- (9) Cluster shade
- (10) Cluster prominence
- (11) Maximum probability
- (12) Sum of squares
- (13) Sum of average
- (14) Sum of variance
- (15) Sum of entropy
- (16) Difference variance
- (17) Difference entropy
- (18) Information correlation criteria (1)
- (19) Information correlation criteria (2)
- (20) Maximum correlation coefficient

- (21) Inverse difference normalized
- (22) Inverse difference moment normalized

*3.2. Machine Learning Classifiers.* One of the learning networks inspired by the perceptron neural network is this kind of neural network. The picture or features linked to the problem are first classified and fed into the grid. The concealed weights in the output layer would then express themselves in several ways [61]. If the output comprises multiple numerical components, the algorithm gives a classification or recognition algorithm (e.g., image classification, normal = 1, abnormal = 2). After many images have been trained, the results are weighted.

The image form is detected when a new image is applied to the algorithm, other than the training images. For instance, the matrix of different images is sent to the method and trained in the computer, such as images of benign or malignant cancers, Alzheimer's, sarcoma, or brain tumors [61]. The process determines the type of disease based on the weights obtained. The convolutional sublayer is the CNN's foundation, and its output matrix can be seen as a 3D matrix of neurons. Standard neural networks are considered for a better understanding. Each layer was a 1D matrix of neurons, producing its output and eventually accumulating a



series of results corresponding to each neuron. While rather than a scalar number, a 3D matrix in which the neurons are arranged in 3D in the CNN is revealed. As a result, this cube's output is also a three-dimensional matrix [61]. The placement of a maxpooling layer among several layers is a widespread technique in traditional architecture. This layer aims to reduce the features and measurements in the network and thus overfit the display, resulting in a smaller input size. The maxpooling feature enlarges or reduces the size of the position. Artificial neural networks' activation mechanism specifies the neuron based on the input matrix [82]. The result values are converted to a goal range, 0 to 1 or -1 to 1 (depending on the activation mechanism used). For instance, the logistic activation function converts all inputs to true absolute ranges [0, 1]. Another way to lose weight is to optimize your weight. In this article, the rectified linear unit (ReLU) for the following functions is employed:

$$f(x) = \begin{cases} 0 & x < 0, \\ x & x \geq 0. \end{cases} \quad (1)$$

The expressions, such as SoftMax, are not exclusive to a single feature and relate to the 1D matrix.

$$f_i(\vec{x}) = \frac{e^{x_i}}{\sum_{j=1}^J e^{x_j}}. \quad (2)$$

In KNN, each tuple in the  $n$ -dimensional space can be called a point if the educational tuples contain  $n$  indexes. The  $k$ -nearest algorithm is specified based on distance measures (e.g., Euclidean distance) in each attribute when a nontuple is given. The test sample is found among the training samples. The test sample's classmark is identical to the labels of the majority of these samples in the test sample's vicinity [83].

SVM is a supervised ML technological category that uses a hyperplane to classify each observation in a given data set. SVM can address linear and nonlinear issues, which is more beneficial in large datasets. SVMs are a generalized linear classifier that can be considered a perceptron extension. They are also known as a particular case of Tikhonov's regularization [84].

Bayesian learning is a computational method for linking data sets to different mathematical approaches by learning conditional independence. Bayesian learning incorporates previous probability functions and new insight to measure later probability. The probability of  $(\theta)$  must be amplified if  $Y_1, Y_2, Y_3, \dots, Y_n$  represents a set of inputs and returns a label. Naive Bayes classifiers can be trained effectively for particular probability models in a supervised learning framework. The maximum likelihood approach is used to estimate parameters for naive Bayes models in many practical applications. In other words, the naive Bayes model [85] can be used without endorsing Bayesian probability or using complex Bayesian techniques.

Fisher's linear discriminant is a technique for determining a linear combination of variables that describe or discriminate two or more sets of objects or occurrences. It is

utilized in statistics and other areas. Fisher's linear discriminant is a generalization of LDA. The resulting mixture may be used as a linear classifier or, more generally, reduce dimensionality before further classification. Discriminant analysis is used when groups are known a priori (unlike in cluster analysis). A score on one or more quantitative predictor variables, as well as a score on a group indicator, is required for each scenario [86]. In basic terms, discriminant function analysis involves grouping, classifying, or categorizing objects into similar groups, classes, or categories.

**3.3. Classifier's Performance Analysis Metrics.** The ROC curve is defined by comparing the true positive rate (TPR) to the false positive rate (FPR). In ML, the TPR is also called recall or probability of detection. Starting on the ROC's left side, the FPR and TPR have vanished. (This implies that the threshold line, which represents the most significant number of test results, is extensive.) Start with the most test results and use that as a starting point. The consistency of the findings of a measure that divides the knowledge into these two categories is measurable and descriptive.

True positive (TP): the classification of the COVID-19 patient is accurate

False positive (FP): the classification of non-COVID patients is with mistakes

True negative (TN): the classification of non-COVID is accurate

False negative (FN): the classification of COVID-19 patients is with mistakes

In mathematical terminology, the sensitivity of separating the percentages of TP cases into the number of TP and FN cases is as follows:

$$\text{Sensitivity} = \frac{TP}{TP + FN}. \quad (3)$$

The function of the techniques listed in artificial intelligence is the confusion matrix. Such a presentation is often utilized in supervised learning algorithms, but it is often employed in unsupervised learning. Each column of the matrix contains an instance of the expected value. Suppose each row has an actual (true) case. Also, the name of this matrix is revealed, allowing for errors and tampering with the outcome [86] (see Figure 1).

## 4. Results and Discussion

**4.1. Dataset.** The dataset includes COVID-19-enhanced X-ray images for disease detection using chest X-ray images. The data were gathered from two publicly accessible datasets [87, 88]. The data are based on a shared open dataset of chest X-rays and CT photographs of patients with COVID-19 or other viral or bacterial pneumonia and are positive or suspicious (MERS, SARS, and ARDS). Data would be compiled directly from hospitals and doctors and indirectly from public records. All photographs and data are made public [46, 50, 53]. Images are in the size of  $256 \times 256$  in PNG format. The example of images is shown in Figure 2. For

analysis and simulation of the methods, 1824 image is used so that 80% belongs to training and 20% is the validation sample. [89].

**4.2. Preprocessing.** In this paper, the dataset of lung X-ray images is collected from the data repository. It converted to a unique PNG format with  $256 \times 256$  pixel size. First, the image should be transformed into a double matrix for analysis. The first and most crucial part of classification in machine learning is feature extraction. Although many deep learning methods use input images to train the network, this paper's GLCM feature extraction method is effective. The critical point in feature extraction methods is that the processing time is reduced, and the size of the output network is highly more minor than the methods with image input. The GLCM approaches extracted 22 features for each image explained in the methods and materials section. The features are normalized between  $[-1, 1]$  in the next step to transform the matrix in an identical range. Two folders of COVID-19 patients and non-COVID patients with 1824 images were finally converted to a matrix with an  $1824 \times 22$  size. Then, images with COVID-19 infections are labeled with (1 or positive) and other images (0 or negative). The final step is the implementation of machine learning methods and extracting performance metrics from the networks. The conceptual diagram of the process is illustrated in Figure 3.

**4.3. Results of Classification.** In this paper, five powerful machine learning techniques are utilized to classify and diagnose the COVID-19 patients with other patients. Five classifiers that analyze and implement diagnosis include KNN, SVM, NB, LDA, and CNN.

Moreover, each class of 912 vectors with 22 features enters the methods. The classification results in a confusion matrix are depicted in Figure 4. Regarding the confusion matrix in Figure 4, orange cells illustrate true metrics, and white cells are false. For example, in KNN results, from 912 images with COVID-19, 825 (45.8% of all images) are detected correctly. However, 77 (4.2% of all images) are misdiagnosed. Therefore, the sensitivity of the KNN methods is 91.6%. In other words, 91.6% of patients with COVID-19 are detected accurately.

Moreover, from 912 images with other diseases, 98.9% are diagnosed correctly. It means that the specificity of the KNN is 98.9%. Furthermore, precision for KNN is 98.8%, which means that 845 patients are detected as COVID-19; however, 835 patients (98.8%) are diagnosed truly. Finally, another important metric for analyzing classifiers is accuracy. It includes all number of true values over the number of true and false values. For the KNN method, the accuracy is 95.2%. It means that from all images, 95.2% ( $902 + 835$ ) of them are diagnosed correctly. Compared with other machine learning methods, the accuracy of KNN is higher than SVM, NB, and LDA approaches. CNN is a feedforward neural network that trains to remove more complex, high-level functionality as neural networks to generate an output map. The convolutional kernel utilizes an input function

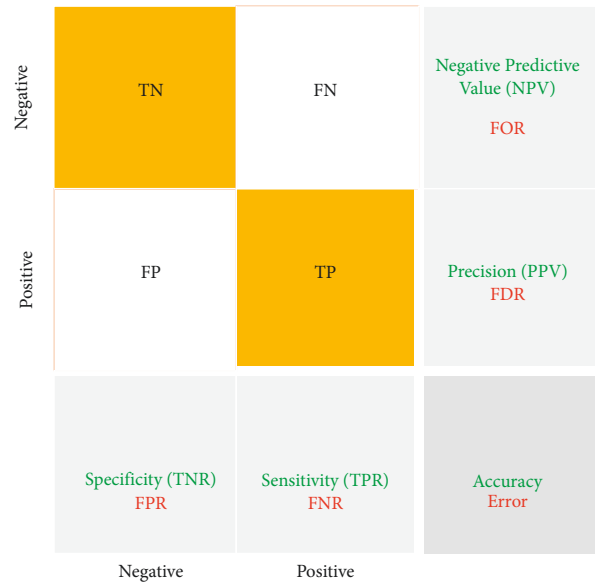


FIGURE 1: The confusion matrix.

map. If a function converts, the output map reflects it, so CNN takes advantage of the fact that a feature is the same in the receptive field no matter where it is. It suggests that CNN can learn more helpful functionality than approaches that do not consider. Because of this assumption, weight sharing is used to minimize the number of factors. The CNNs learn through gradient descent. Each layer feeds into the one below it, resulting in hierarchical features tailored to the task at hand. Features in the form of a real-valued vector are typically required by SVM and others.

On the other hand, CNN is typically taught from beginning to end, ensuring it responds to the dilemma it is trying to solve. SVMs, KNNs, and random forests use CNN as a trainable attribute detector. Different algorithms in machine learning should be complementary since SVMs are still widely used; it only depends on the task. Therefore, this paper uses the CNN method to classify COVID-19 patients using GLCM features. The structure of CNN architecture is shown in Figure 5.

The input layer consists of 22 GLCM normalized features. It is used as a vector for the simulation of the network. Finally, the categorical output labels are 1 for positive samples (COVID-19 patients) and 0 for negative samples (other patients). The simulation has PC core i7, 2.5 GHz, and 12 GB RAM. The process of training is shown in Figure 6. The process has continued to 5800 iterations to reach an acceptable accuracy. The complexity of SVM is  $O(n^3)$ , KNN  $O(n dk)$ , NB  $O(n d)$ , LDA  $O(mn \cdot \min(m, n))$ , CNN  $O(n)$ .

The confusion matrix of the CNN results is illustrated in Figure 7. Based on the matrix, from 912 COVID-19 patients, 905 (99.2%) were diagnosed correctly. However, only seven images are misdiagnosed. On the other hand, all the patients with other diseases or negative results are detected. In other words, the sensitivity and specificity of the CNN methods are 99.2% and 100%. Moreover, the precision is 100%. It means that all the patients detected as COVID-19 are truly correct.

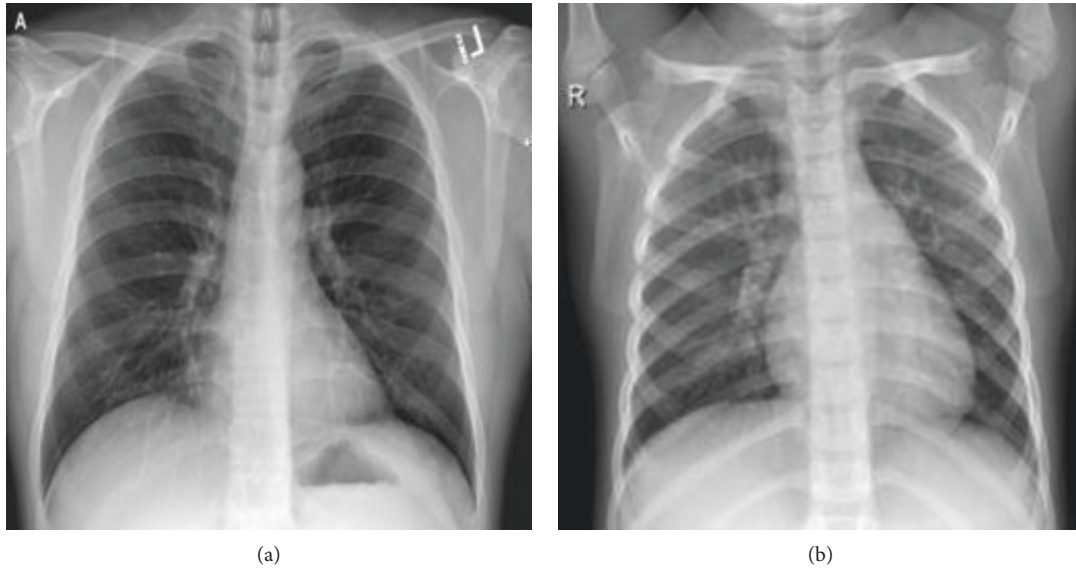


FIGURE 2: Example of X-ray image from patients' lungs. (a) COVID-19 patient. (b) Other patients.

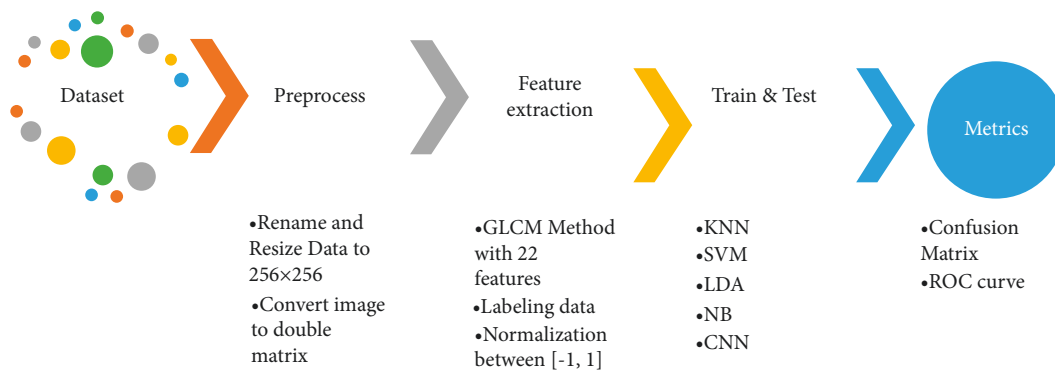


FIGURE 3: Conceptual diagram of the presented method.

Consequently, the accuracy of the CNN methods is 99.6%. For comparison of the used machine learning methods, the ROC curve is depicted in Figure 8. Regarding the ROC curve, the horizontal axis is fallout or FPR, and the vertical axis is sensitivity or TPR. If the fallout and sensitivity are lower and higher values, respectively, it is desirable for classification. Therefore, CNN methods result in higher sensitivity in comparison with other methods. Also, NB illustrated lower sensitivity between machine learning methods.

Based on the performance analysis metrics results shown in Table 2, the higher values belong to the CNN technique. Under the ROC curve value, AUC is another essential metric for classifiers. For CNN methods, it is 99.97. Consequently, the highest accuracy belongs to CNN, KNN, LDA, SVM, and NB, respectively.

### 5. Discussion

As medical image processing technology has progressed, intelligent detection and diagnosis software have accelerated. A vital tool for boosting diagnosis accuracy is machine learning algorithms, widely regarded as adequate. However, to generate superior machine learning models, it is vital to use effective feature extraction methods. Therefore, deep learning models are often utilized in medical imaging applications because they can automatically retrieve features or employ pretrained models. As a consequence, deep learning models are becoming increasingly popular. The findings of this article, which employs machine learning algorithms to identify COVID-19 on chest X-rays, are extremely encouraging in their preliminary nature. GLCM algorithms are used to extract the features from the data.

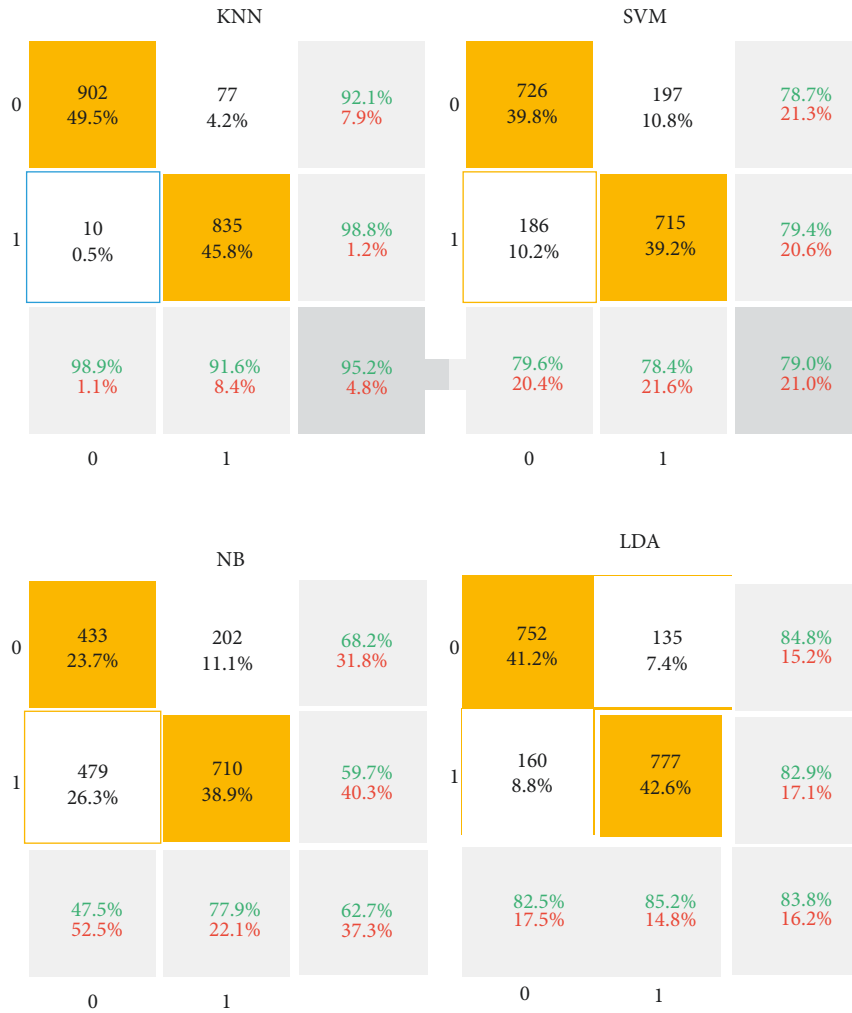


FIGURE 4: The confusion matrix of the deep learning methods used for COVID-19 diagnosis.

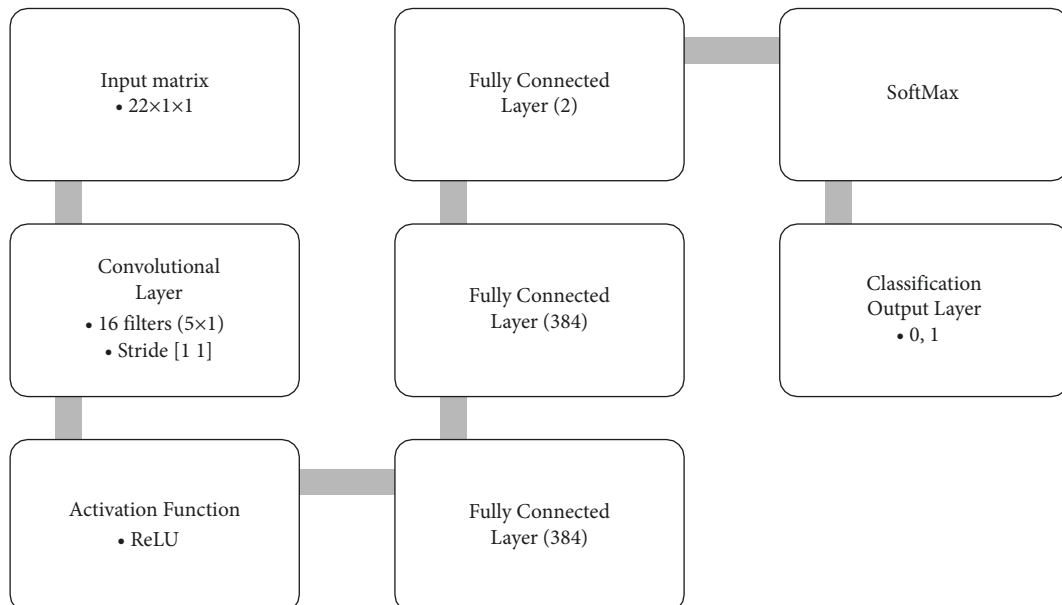


FIGURE 5: The architecture of presented CNN methods for features classification.

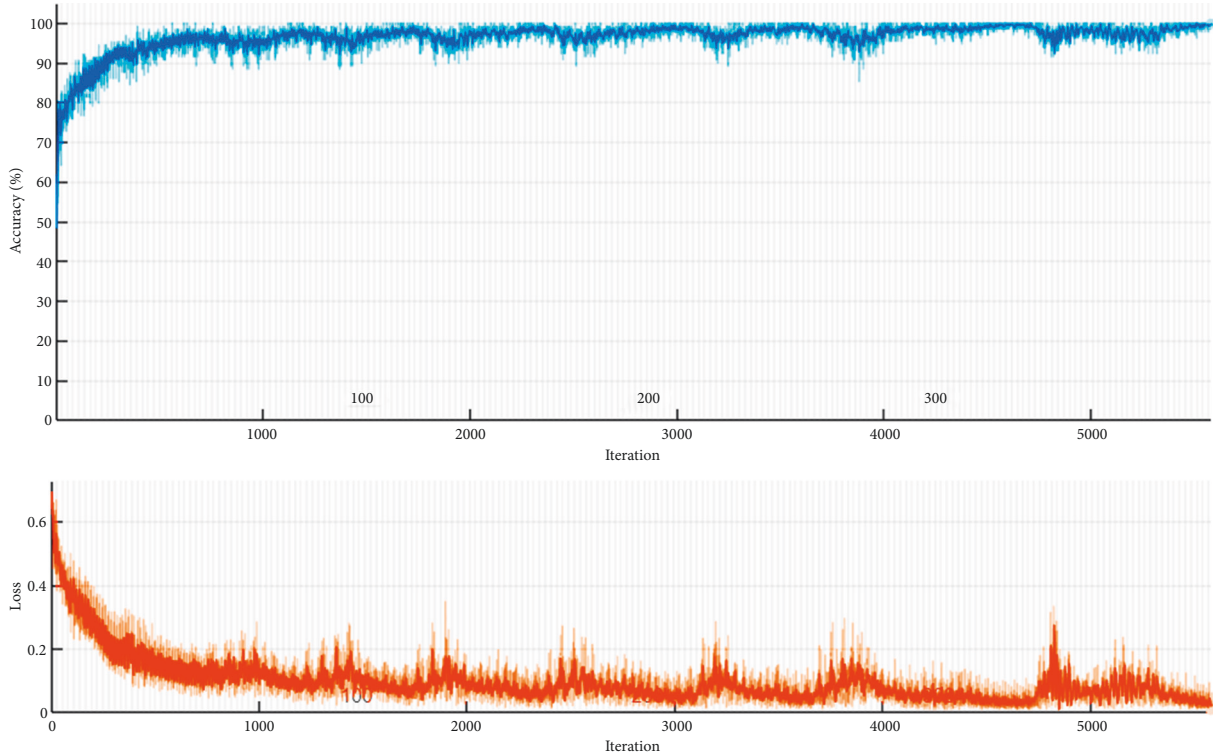


FIGURE 6: The training process of the CNN approach.

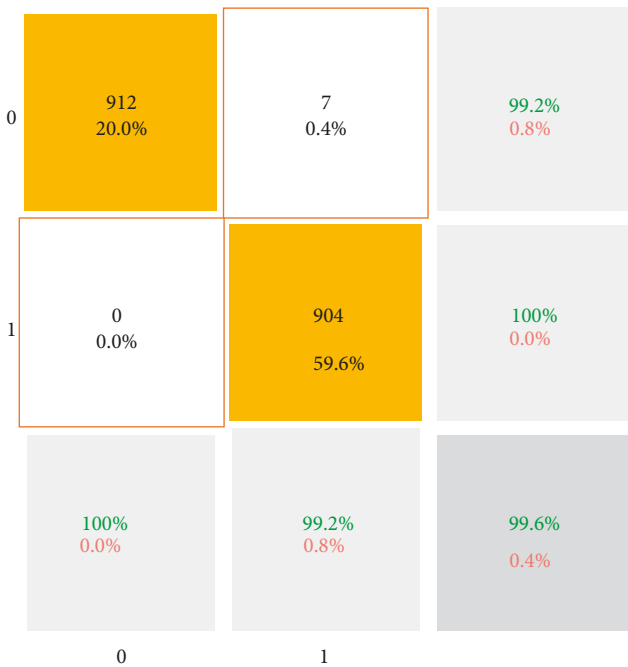


FIGURE 7: The confusion matrix of the presented CNN method.

The followings are the broad thumb guidelines employed in selecting textural features: because the values of energy are within a normalized range, it is chosen over entropy. The average gray level difference between neighboring pixels is connected with contrast. It is similar to variance, but it is chosen because of the lower computing effort and usefulness

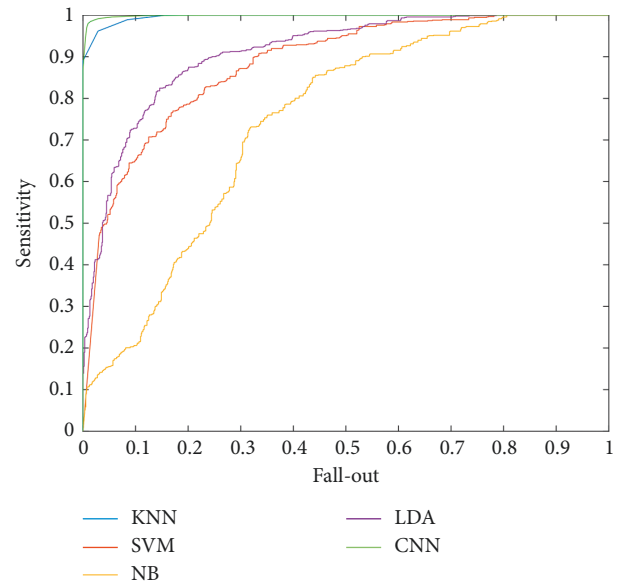


FIGURE 8: The ROC curve for presented methods.

as a spatial frequency measure. In addition, energy and contrast are essential characteristics in terms of visual evaluation and computational burden to discern between distinct textural patterns. The categorization of patients is accomplished using KNN, SVM, LDA, NB, and CNN algorithms. It would be beneficial to have a large amount of memory and computer capacity for working with many variables or require an algorithm to categorize the exhibits and apply them to fresh samples. Feature extraction is a

TABLE 2: The comparison between the presented methods.

Methods	Sensitivity (%)	Specificity (%)	Precision (%)	AUC	Accuracy (%)
KNN	91.6	98.9	98.8	99.61	95.2
SVM	78.4	76.6	79.4	88.20	79.0
NB	77.9	47.5	59.7	74.24	62.7
LDA	85.2	82.5	82.9	90.94	83.8
CNN	99.2	100	100	99.97	99.6
[14]	96.1	99.7	96.9	—	93.2

broad idea that may be used to build dynamic mixes of variables that can solve these challenges with pinpoint precision. To achieve accurate object recognition and division, texture analysis tried to discover a reliable technique to express the distinctive attributes of textures in a simplified yet specific manner for object detection and division. The surface is critical in the interpretation of images and the recognition of patterns. Texture extraction is only employed in a few CPU architectures, and it is not widely available.

Additionally, the accuracy is excellent. It follows that all of the patients who were previously diagnosed as having COVID-19 were correct in their diagnosis. Consequently, the best accuracy is achieved by CNN, KNN, LDA, SVM, and NB, in that order. By categorizing lungs and adding information paired with lungs around noise, there is a significant likelihood of supplying more data, resulting in enhanced outcomes. This disturbance is linked to cables and collected equipment, patients' ages, and gender, allowing photos without lungs to be classified precisely. Due to noise bias, future applications utilizing models without lungs may have an increased risk of mislabeling pictures. To verify that noise is not a cause of bias, more work is needed to segregate diseases diagnosed by the expert radiologist. It is also important to note that the findings reported do not always imply the same performance across all datasets. Principal datasets, for example, originate from European patients; other global patients may have modest data capture differences or diseases, implying that a better categorization utilizing global datasets is required. Separating the datasets by gender will also offer further information on the model's scope, as the soft tissues of the breast might conceal sections of the lungs. It is unclear whether this represents a bias in the model's prediction.

## 6. Conclusion

The growth of intelligent detection and diagnostic software has accelerated as medical image processing has advanced. Machine learning algorithms are commonly recognized as a powerful method for improving disease diagnostic accuracy. However, efficient feature extraction tools are necessary to obtain better machine learning models. As a result, deep learning models are commonly used in medical imaging applications because they can automatically retrieve features or use pretrained models. This paper uses machine learning methods to diagnose COVID-19 on chest X-rays, and the findings have been very promising. The feature extractions are done using GLCM methods. For the classification of patients, KNN, SVM, LDA, NB, and CNN methods are used.

The sensitivity of the KNN methods is 91.6 percent, based on the findings. In other words, 91.6 percent of COVID-19 patients are correctly identified. Furthermore, when comparing the process 912 images to other diseases, 98.9% of the disorder is correctly identified. It means that the KNN has a specificity of 98.9%. Furthermore, KNN has a precision of 98.8%, which indicates that 845 patients are observed as having COVID-19 disorder, but only 835 (98.8%) are diagnosed. It means that 95.2 percent of the images are correctly diagnosed. In contrast to other machine learning techniques, the precision of KNN is better than that of SVM, NB, and LDA. Based on the index, 905 (99.2%) of 912 COVID-19 patients were correctly diagnosed. Only seven images, however, are misdiagnosed. Patients with other diseases or unfavorable effects, on the other hand, are identified. In other words, the CNN approaches have a sensitivity and specificity of 99.2 percent and 100 percent, respectively. Furthermore, the precision is fine. It means that all of the patients who were identified as having COVID-19 are right. As a result, CNN, KNN, LDA, SVM, and NB have the best accuracy, respectively. Future works should extend feature-based methods for classifying diseases such as COVID-19, Alzheimer's, and lung cancer instead of image base method. It prevents time constraints and develops fast methods [90, 91].

## Data Availability

Data are available and can be provided over the emails querying directly to the corresponding author (f.abolhasani@kmu.ac.ir).

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] T. Ai, Z. Yang, H. Hou et al., "Correlation of chest CT and RT-PCR testing in coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases," *Radiology*, vol. 296, Article ID 200642, 2020.
- [2] Y. Fang, H. Zhang, J. Xie et al., "Sensitivity of chest CT for COVID-19: comparison to RT-PCR," *Radiology*, vol. 296, no. 2, pp. E115–E117, 2020.
- [3] H. Zeng, C. Xu, J. Fan et al., "Antibodies in infants born to mothers with COVID-19 pneumonia," *JAMA*, vol. 18, pp. 1848–1849, 2020.
- [4] D. A. Schwartz, "An analysis of 38 pregnant women with COVID-19, their newborn infants, and maternal-fetal transmission of SARS-CoV-2: maternal coronavirus

- infections and pregnancy outcomes,” *Archives of Pathology & Laboratory Medicine*, vol. 144, no. 7, pp. 799–805, 2020.
- [5] T. Ozturk, M. Talo, E. A. Yildirim, U. B. Baloglu, O. Yildirim, and U. Rajendra Acharya, “Automated detection of COVID-19 cases using deep neural networks with X-ray images,” *Computers in Biology and Medicine*, vol. 121, Article ID 103792, 2020.
  - [6] M. Li, P. Lei, B. Zeng et al., “Coronavirus disease (COVID-19): spectrum of CT findings and temporal progression of the disease,” *Academic Radiology*, vol. 27, no. 5, pp. 603–608, 2020.
  - [7] F. Shi, J. Wang, J. Shi, and W. Ziyang, “Review of artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for covid-19,” *IEEE Rev Biomed Eng*, vol. 14, pp. 1-2, 2020.
  - [8] H. S. Maghddid, A. T. Asaad, K. Z. Ghafoor, A. S. Sadiq, and M. K. Khan, “Diagnosing COVID-19 pneumonia from X-ray and CT images using deep learning and transfer learning algorithms,” 2020, <https://arxiv.org/abs/2004.00038>.
  - [9] A. F. de Moraes Batista, J. L. Miraglia, T. H. R. Donato, and A. D. P. C. Filho, “COVID-19 diagnosis prediction in emergency care patients: a machine learning approach,” medRxiv, <https://www.medrxiv.org/content/10.1101/2020.04.04.20052092v2>, 2020.
  - [10] G. Hripcsak and D. J. Albers, “Next-generation phenotyping of electronic health records,” *Journal of the American Medical Informatics Association*, vol. 20, no. 1, pp. 117–121, 2013.
  - [11] P. B. Jensen, L. J. Jensen, and S. Brunak, “Mining electronic health records: towards better research applications and clinical care,” *Nature Reviews Genetics*, vol. 13, no. 6, pp. 395–405, 2012.
  - [12] J. Luo, M. Gopukumar, D. Zhao, and W. Min, “Big data application in biomedical research and health care: a literature review,” *Biomedical Informatics Insights*, vol. 8, Article ID BII.S31559, 2016.
  - [13] M. Ahmadi, A. Sharifi, and S. Khalili, “Presentation of a developed sub-epidemic model for estimation of the COVID-19 pandemic and assessment of travel-related risks in Iran,” *Environmental Science and Pollution Research*, vol. 28, no. 12, Article ID 14521, 2020.
  - [14] S. Hassantabar, M. Ahmadi, and A. Sharifi, “Diagnosis and detection of infected tissue of COVID-19 patients based on lung X-ray image using convolutional neural network approaches,” *Chaos, Solitons & Fractals*, vol. 140, Article ID 110170, 2020.
  - [15] M. H. Nasirpour, A. Sharifi, M. Ahmadi, and S. Jafarzadeh Ghouschi, “Revealing the relationship between solar activity and COVID-19 and forecasting of possible future viruses using multi-step autoregression (MSAR),” *Environmental Science and Pollution Research*, vol. 28, no. 28, Article ID 38074, 2021.
  - [16] P. K. Chaudhary and R. B. Pachori, “FBSED based automatic diagnosis of COVID-19 using X-ray and CT images,” *Computers in Biology and Medicine*, vol. 134, Article ID 104454, 2021.
  - [17] M. Q. H. Abadi, S. Rahmati, A. Sharifi, and M. Ahmadi, “HSSAGA: designation and scheduling of nurses for taking care of COVID-19 patients using novel method of Hybrid Salp Swarm Algorithm and Genetic Algorithm,” *Applied Soft Computing*, vol. 108, Article ID 107449, 2021.
  - [18] A. Davoudi, M. Ahmadi, A. Sharifi et al., “Studying the effect of taking statins before infection in the severity reduction of COVID-19 with machine learning,” *BioMed Research International*, vol. 202112 pages, Article ID 9995073, 2021.
  - [19] M. F. Aslan, M. F. Unlarsen, K. Sabanci, and A. Durdu, “CNN-based transfer learning–BiLSTM network: a novel approach for COVID-19 infection detection,” *Applied Soft Computing*, vol. 98, Article ID 106912, 2021.
  - [20] G. Jia, H. K. Lam, and Y. Xu, “Classification of COVID-19 chest X-Ray and CT images using a type of dynamic CNN modification method,” *Computers in Biology and Medicine*, vol. 134, Article ID 104425, 2021.
  - [21] M. M. Islam, F. Karray, R. Alhaji, and J. Zeng, “A review on deep learning techniques for the diagnosis of novel coronavirus (covid-19),” *IEEE Access*, vol. 9, Article ID 30551, 2021.
  - [22] M. Wainberg, D. Merico, A. Delong, and B. J. Frey, “Deep learning in biomedicine,” *Nature Biotechnology*, vol. 36, no. 9, pp. 829–838, 2018.
  - [23] A. Esteva, A. Robicquet, B. Ramsundar et al., “A guide to deep learning in healthcare,” *Nature Medicine*, vol. 25, no. 1, pp. 24–29, 2019.
  - [24] D. Shen, G. Wu, and H.-I. Suk, “Deep learning in medical image analysis,” *Annual Review of Biomedical Engineering*, vol. 19, no. 1, pp. 221–248, 2017.
  - [25] H. Chen, O. Engkvist, Y. Wang, M. Olivecrona, and T. Blaschke, “The rise of deep learning in drug discovery,” *Drug Discovery Today*, vol. 23, no. 6, pp. 1241–1250, 2018.
  - [26] E. Eslami and H.-B. Yun, “Attention-based multi-scale convolutional neural network (A+MCNN) for multi-class classification in road images,” *Sensors*, vol. 21, no. 15, p. 5137, 2021.
  - [27] G. Litjens, T. Kooi, B. E. Bejnordi et al., “A survey on deep learning in medical image analysis,” *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.
  - [28] A. Elhassouny and F. Smarandache, “Trends in deep convolutional neural networks architectures: a review,” in *Proceedings of the International Conference of Computer Science and Renewable Energies (ICCSRE)*, pp. 1–8, Agadir, Morocco, July 2019.
  - [29] Q. Xu, Y. Zeng, W. Tang et al., “Multi-task Joint learning model for segmenting and classifying Tongue images using a deep neural network,” *IEEE journal of biomedical and health informatics*, vol. 24, no. 9, pp. 2481–2489, 2020.
  - [30] Y. Zou, H. Wu, X. Guo et al., “MK-FSVM-SVDD: a multiple kernel-based fuzzy SVM model for predicting DNA-binding proteins via support vector data description,” *Current Bioinformatics*, vol. 16, no. 2, pp. 274–283, 2021.
  - [31] M. Z. Alom, T. M. Taha, C. Yakopcic et al., “A state-of-the-art survey on deep learning theory and architectures,” *Electronics*, vol. 8, no. 3, 2019.
  - [32] S. Das, “CNN architectures: LeNet, AlexNet, VGG, GoogleLeNet, ResNet and more,” 2017, <https://medium.com/analyticsvidhya/cnns-architectures-lexnet-alexnet-vgg-googlenet-resnet-andmore-666091488df5>.
  - [33] G. Zeng, Y. He, Z. Yu, X. Yang, R. Yang, and L. Zhang, “Preparation of novel high copper ions removal membranes by embedding organosilane-functionalized multi-walled carbon nanotube,” *Journal of Chemical Technology & Biotechnology*, vol. 91, no. 8, pp. 2322–2330, 2016.
  - [34] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size,” 2016, <http://arxiv.org/abs/1602.07360>.
  - [35] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014, <http://arxiv.org/abs/1409.1556>.
  - [36] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” 2015, <http://arxiv.org/abs/1512.03385>.

- [37] F. Chollet, "Xception: deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1800–1807, Honolulu, HI, USA, July 2017.
- [38] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 4278–4284, California, CL, USA, February 2017.
- [39] Z. Qin, Z. Zhang, X. Chen, C. Wang, and Y. Peng, "Fdmobilenet: improved mobilenet with a fast downsampling strategy," in *Proceedings of the 25th IEEE International Conference on Image Processing (ICIP)*, pp. 1363–1367, Athens, Greece, October 2018.
- [40] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2261–2269, Honolulu, HI, USA, July 2017.
- [41] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: convolutional networks for biomedical image segmentation," *Lecture Notes in Computer Science*, in *Proceedings of the International Conference. Med. Image Comput. Comput.-Assist. Intervent*, pp. 234–241, 2015.
- [42] L. Li, L. Qin, Z. Xu et al., "Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT," *Radiology*, vol. 19, Article ID 200905, 2020.
- [43] X. Wu, H. Hui, M. Niu et al., "Deep learning-based multi-view fusion model for screening 2019 novel coronavirus pneumonia: a multicentre study," *European Journal of Radiology*, vol. 128, Article ID 109041, 2020.
- [44] A. Rezaei, H. Hassani, P. Moarefvand, and A. Golmohammadi, "Lithological mapping in Sangan region in Northeast Iran using ASTER satellite data and image processing methods," *Geology, Ecology, and Landscapes*, vol. 4, no. 1, pp. 59–70, 2020.
- [45] A. Arenliu, F. Uka, and S. Weine, "Building online and telephone psychological first aid services in a low resource setting during covid-19: the case of kosovo," *Psychiatry Danubina*, vol. 32, no. 3-4, pp. 570–576, 2020.
- [46] M. Rezaei, F. Farahanipad, A. Dillhoff, R. Elmasri, and V. Athitsos, "Weakly-supervised hand part segmentation from depth images," in *Proceedings of the 14th PErvasive Technologies Related to Assistive Environments Conference*, pp. 218–225, New York, NY, USA, 2021, June.
- [47] M. Yousefzadeh, P. Esfahanian, and S. Movahed, "Ai-corona: radiologist-assistant deep learning framework for COVID-19 diagnosis in chest ct scans," *MedRxiv*, 2020, <https://www.medrxiv.org/content/10.1101/2020.05.04.20082081v1>.
- [48] C. Jin, W. Chen, Y. Cao et al., "Development and evaluation of an artificial intelligence system for COVID-19 diagnosis," *Nature Communications*, vol. 11, no. 1, 2020.
- [49] S. Rehman and U. Lela, "Psychological aid to covid-19 pandemic: a mental health response to crises management," *Psychiatry Danubina*, vol. 32, no. 2, pp. 262–265, 2020.
- [50] A. Radulescu and K. Cavanagh, "Management strategies in a SEIR model of COVID 19 community spread," 2020, <https://arxiv.org/abs/2003.11150>.
- [51] R. Gupta, "SEIR and Regression Model based COVID- 19 outbreak predictions in India," *medRxiv*, 2020, <https://www.medrxiv.org/content/10.1101/2020.04.01.20049825v1>.
- [52] M. Zhang, Y. Chen, and W. Susilo, "PPO-CPQ: a privacy-preserving optimization of clinical pathway query for E-healthcare systems," *IEEE Internet of Things Journal*, vol. 7, no. 10, Article ID 10660, 2020.
- [53] H. Liu, J. Liu, S. Hou, T. Tao, and J. Han, "Perception consistency ultrasound image super-resolution via self-supervised CycleGAN," *Neural Computing & Applications*, 2021.
- [54] M. Mahmoudi, "COVID Lessons: was there any way to reduce the negative effect of COVID-19 on the United States economy?," 2022, <https://arxiv.org/abs/2201.00274>.
- [55] K. Prem, Y. Liu, and T. Russell, "The effect of control strategies to reduce social mixing on outcomes of the COVID-19 epidemic in Wuhan, China: a modelling study," *The Lancet Public Health*, vol. 5, 2020.
- [56] H. Zhang, X. Guo, and Y. Zeng, *Transmissibility of COVID-19 and its Association with Temperature and Humidity*, 2020.
- [57] Z. A. A. Alyasseri, M. A. AlBetar, I. A. Doush et al., "Review on COVID-19 diagnosis models based on machine learning and deep learning approaches," *Expert Systems*, Article ID e12759, 2021.
- [58] L. Wang and A. Wong, "COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest radiography images," 2020, <https://arxiv.org/abs/2003.09871>.
- [59] A. Narin, C. Kaya, and Z. Pamuk, "Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks," 2020, <https://arxiv.org/abs/2003.10849>.
- [60] L. Li, Q. Lixin, X. Zeguo, Y. Youbing, and W. Xin, "Artificial intelligence distinguishes covid-19 from community acquired pneumonia on chest ct," in *Radiology*, Article ID 200905, 2020.
- [61] I. D. Apostolopoulos and T. A. Mpesiana, "Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks," *Physical and Engineering Sciences in Medicine*, vol. 43, 2020.
- [62] E. E.-D. Hemdan, M. A. Shouman, and M. E. Karar, "Covidxnet: a framework of deep learning classifiers to diagnose covid-19 in x-ray images," 2020, <https://arxiv.org/abs/2003.11055>.
- [63] M. Barstugan, U. Ozkaya, and S. Ozturk, "Coronavirus (covid- 19) classification using ct images by machine learning methods," 2020, <https://arxiv.org/abs/2003.09424>.
- [64] P. K. Sethy and S. K. Behera, "Detection of coronavirus disease (covid-19) based on deep features and support vector machine," *Int. J. Math. Eng. Mana. Sci.*, vol. 5, 2020.
- [65] A. F. de Moraes Batista, J. L. Miraglia, T. H. R. Donato, and A. D. P. C. Filho, "COVID-19 diagnosis prediction in emergency care patients: a machine learning approach," *medRxiv*, 2020, <https://www.medrxiv.org/content/10.1101/2020.04.04.20052092v2>.
- [66] A. E. Hassanien, L. N. Mahdy, K. A. Ezzat, and H. A. Ella, "Automatic x-ray COVID-19 lung image classification system based on multi-level thresholding and support vector machine," *medRxiv*, 2020, <https://www.medrxiv.org/content/10.1101/2020.03.30.20047787v1>.
- [67] K. Li, J. Wu, F. Wu, D. Guo, and L. Chen, "The clinical and chest CT features associated with severe and critical COVID-19 pneumonia," *Investigative Radiology*, vol. 55, 2020.
- [68] R. Kumar, R. Arora, V. Bansal, H. Buckchash, and J. Imran, "Accurate prediction of COVID-19 using chest xray images through deep feature learning model with smote and machine learning classifiers," *medRxiv*, 2020, <https://www.medrxiv.org/content/10.1101/2020.04.13.20063461v1>.
- [69] A. A. Farid, G. I. Selim, and H. A. A. Khater, "A novel approach of CT images feature analysis and prediction to screen



- for corona virus disease (COVID-19),” *International Journal of Scientific Engineering and Research*, vol. 11, no. 3, pp. 1141–1149, 2020.
- [70] C. Li, X. Xiao, T. Li, X. Ding, J. Xiao, and W. Fu, “Preliminary study to identify severe from moderate cases of COVID-19 using NLR&RDW-SD combination parameter,” *medRxiv*, 2020, <https://www.medrxiv.org/content/10.1101/2020.04.09.20058594v1>.
- [71] Z. Tang, W. Zhao, X. Xie, and Z. Zhong, “Severity assessment of coronavirus disease 2019 (COVID-19) using quantitative features from chest CT images,” 2020, <https://arxiv.org/abs/2003.11988>.
- [72] F. Shi, L. Xia, F. Shan, D. Wu, and Y. Wei, “Large-scale screening of covid-19 from community acquired pneumonia using infection size-aware classification,” 2020, <https://arxiv.org/abs/2003.09860>.
- [73] X. Chen, L. Yao, and Y. Zhang, “Residual attention U-net for automated multi-class segmentation of COVID-19 chest CT images,” 2020, <https://arxiv.org/abs/2004.05645>.
- [74] G. Gaal, B. Maga, and A. Luk’acs, “Attention U-net based adversarial architectures for chest X-ray lung segmentation,” 2020, <https://arxiv.org/abs/2003.10304>.
- [75] D.-P. Fan, T. Zhou, G.-P. Ji, Y. Zhou, and G. Chen, “Inf-Net: automatic COVID-19 lung infection segmentation from CT scans,” 2020, <https://arxiv.org/abs/2004.14133>.
- [76] V. Kumar, D. Singh, M. Kaur, and R. Damaševičius, “Overview of current state of research on the application of artificial intelligence techniques for COVID-19,” *PeerJ computer science*, vol. 7, 2021.
- [77] T. Akram, M. Attique, S. Gul et al., “A novel framework for rapid diagnosis of COVID-19 on computed tomography scans,” *Pattern Analysis & Applications*, vol. 24, no. 3, pp. 951–964, 2021.
- [78] M. A. Khan, M. Alhaisoni, U. Tariq et al., “COVID-19 case recognition from chest CT images by deep learning, entropy-controlled firefly optimization, and parallel feature fusion,” *Sensors*, vol. 21, no. 21, 2021.
- [79] N. Nejatishahidin, P. Fayyazsanavi, and J. Kosecka, “Object pose estimation using mid-level visual representations,” 2020, <https://arxiv.org/abs/2203.01449>.
- [80] N. U. Rehman, M. S. Zia, T. Meraj et al., “A self-activated cnn approach for multi-class chest-related COVID-19 detection,” *Applied Sciences*, vol. 11, no. 19, 2021.
- [81] D. A. Clausi, “An analysis of co-occurrence texture statistics as a function of grey level quantization,” *Canadian Journal of Remote Sensing*, vol. 28, no. 1, pp. 45–62, 2002.
- [82] M. Ahmadi, A. Sharifi, M. Jafarian Fard, and N. Soleimani, “Detection of brain lesion location in MRI images using convolutional neural network and robust PCA,” *International Journal of Neuroscience*, vol. 12, pp. 1–12, 2021.
- [83] J. M. Keller, M. R. Gray, and J. A. Givens, “A fuzzy k-nearest neighbor algorithm,” *IEEE transactions on systems, man, and cybernetics*, vol. SMC-15, no. 4, pp. 580–585, 1985.
- [84] D. Meyer, F. Leisch, and K. Hornik, “The support vector machine under test,” *Neurocomputing*, vol. 55, no. 1–2, pp. 169–186, 2003.
- [85] A. McCallum, *Graphical Models, Lecture2: Bayesian Network Representation*, 2019.
- [86] A. Sharifi, M. Ahmadi, M. A. Mehni, S. Jafarzadeh Ghoushchi, and Y. Pourasad, “Experimental and numerical diagnosis of fatigue foot using convolutional neural network,” *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 24, no. 16, pp. 1828–1840, 2021.
- [87] P. C. Joseph, M. Paul, D. Lan, R. Karsten, Q. D. Tim, and G. Marzyeh, “COVID-19 image data collection: prospective predictions are the future,” 2020, <https://github.com/ieee8023/covid-chestxray-dataset>.
- [88] D. S. Kermany, M. Goldbaum, W. Cai et al., “Identifying medical diagnoses and treatable diseases by image-based deep learning,” *Cell*, vol. 172, no. 5, pp. 1122–1131.e9, 2018.
- [89] M. Parra Gordo, G. Buitrago Weiland, M. Grau García, and G. Arenaza Choperena, “Radiologic aspects of COVID-19 pneumonia: outcomes and thoracic complications,” *Radiología*, vol. 63, no. 1, pp. 74–88, 2021.
- [90] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning—An MIT Press Book*, MIT Press, Cambridge, MA, USA, 2016.
- [91] S. E. Eikenberry, M. Mancuso, E. Iboi, and T. Phan, “To mask or not to mask: modeling the potential for face mask use by the general public to curtail the COVID-19 pandemic,” *Infectious disease modelling*, vol. 5, 2020.

## Research Article

# 5G Converged Network Resource Allocation Strategy Based on Reinforcement Learning in Edge Cloud Computing Environment

Xuezhu Li 

*Department of Information Engineering, Suzhou University, Suzhou, Anhui 234000, China*

Correspondence should be addressed to Xuezhu Li; [lixuezhu019@163.com](mailto:lixuezhu019@163.com)

Received 21 January 2022; Revised 23 March 2022; Accepted 25 April 2022; Published 14 May 2022

Academic Editor: Laith Abualigah

Copyright © 2022 Xuezhu Li. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Aiming at the problem that computing power and resources of Mobile Edge Computing (MEC) servers are difficult to process long-period intensive task data, this study proposes a 5G converged network resource allocation strategy based on reinforcement learning in edge cloud computing environment. In order to solve the problem of insufficient local computing power, the proposed strategy offloads some tasks to the edge of network. Firstly, we build a multi-MEC server and multi-user mobile edge system, and design optimization objectives to minimize the average response time of system tasks and total energy consumption. Then, task offloading and resource allocation process is modeled as Markov decision process. Furthermore, the deep Q-network is used to find the optimal resource allocation scheme. Finally, the proposed strategy is analyzed experimentally based on TensorFlow learning framework. Experimental results show that when the number of users is 110, final energy consumption is about 2500 J, which effectively reduces task delay and improves the utilization of resources.

## 1. Introduction

With the continuous development of technologies such as the 5G, the amount of data in various emerging application scenarios has exponentially increased. There are more and more Internet of Things (IoT) devices in various fields such as telemedicine, smart car driving, and smart cities, so all kinds of computing are everywhere [1]. However, the existing cloud computing models are difficult to manage these large-scale computing resources and perform data analysis. This is mainly reflected in the following two reasons: First, the transfer of large-scale data to cloud computing center will improve network performance and the computing power of cloud computing infrastructure brings severe challenges [2,3]. The second is that it is difficult for cloud far away from users to meet the stringent requirements of new applications such as autonomous driving on network delay and response speed [4]. Thus, both computing services and big data sources are undergoing a shift from cloud to edge [5].

Edge computing serves as an intermediate layer between the cloud computing center and user devices. It provides computing resources to users near the edge via a high-speed

network by placing the edge server close to user end [6]. Among them, the user device sends computing tasks that originally need to be sent to the cloud or executed locally to edge server for execution to achieve reasonable network resource allocation, which is called computing offloading [7]. Compared with cloud servers and local computing, edge computing can provide faster network response and have more powerful computing capabilities [8]. Therefore, computing offloading and reasonable allocation of network resources by a reasonable scheduling algorithm can help users save transmission energy consumption and improve computing efficiency [9].

In the edge computing system, for security and efficiency reasons, the edge server will not open its own computing resource configuration and idle state to each user device, so it is difficult to obtain the detailed status in the system [10,11]. Under the premise of incomplete observation system constraints, task offloading and system optimization problems become more complicated. The intelligent model represented by deep reinforcement learning is an important means to solve such problems [12]. Reference [13] developed a Multi-Agent Reinforcement Learning Network to solve the Q-learning problem based on independent learners, and

designed the calculation unloading strategy in IoT through random game. However, the efficiency of resource allocation strategy needs to be further improved. Reference [14] proposed a moving edge computing (MEC) network based on blockchain, which uses blockchain to control the coverage system, and adopts adaptive strategy to generate blocks and realize high-quality resource allocation. Reference [15] used deep Q network (DQN) learning to obtain the best resource allocation scheme in IoT network. However, frequent data interaction brings high network load, which becomes the main obstacle to the training of intelligent offloading models, especially computing offloading models based on deep learning.

Traditional methods also have certain research on computing task offloading and network resource allocation: For example, reference [16] solved the task unloading problem based on differential evolution algorithm, so as to realize the efficient execution of tasks, but it requires higher network bandwidth. Reference [17] designed a random mixed integer nonlinear programming method for the intensive task unloading and resource allocation in MEC, which can realize the rational use of resources, but cannot take into account energy efficiency and service delay. Reference [18] used orthogonal and non-orthogonal multiple access methods, a resource allocation scheme considering energy consumption and efficiency in MEC is formulated, but the overall delay needs to be further reduced. Reference [19] proposed a multi-objective resource allocation method for MEC, which uses Pareto archiving evolution strategy to optimize time cost and load balancing. At the same time, it combined multi-criteria decision-making and sorting preference technology similar to the ideal solution to obtain optimal resource allocation, but 5G integration scheme is not considered.

Aiming at the problem that the large amount of data transmitted in 5G network leads to channel congestion, which affects the real-time performance and energy consumption of communication, a 5G fusion network resource allocation strategy based on reinforcement learning in edge cloud computing environment is proposed. Due to the poor learning effect of basic reinforcement learning in massive data, the proposed strategy proposes a DQN offloading strategy to solve resource allocation of 5G converged networks, which can reduce the time delay. At the same time, the system energy consumption is reduced. Finally, experimental results based on TensorFlow learning framework show that proposed strategy fully considers the time and energy consumption of local and offloading to MEC execution, and solving offloading scheme by reinforcement learning can greatly reduce delay and energy consumption. Moreover, its energy consumption is about 2500J, the time delay does not exceed 7s. DQN has self-learning ability, which continuously learns during the training process to improve the accuracy of decision-making. Therefore, it can effectively reduce load and broadband utilization rate.

## 2. System Scenario and Optimization Goal

*2.1. System Scenario.* The system scenario is shown in Figure 1, consisting of  $N$  users,  $M$  base stations, and multiple

MEC servers. Among them, each user is associated with the nearest base station through the wireless link and sends a task request to it. At the same time, each base station is equipped with an MEC server with multiple CPU cores. Therefore, MEC server can process the computing tasks of different users in parallel. It is assumed that user computing tasks are processed by an MEC server, regardless of situation in which computing tasks are forwarded between MEC servers.

Divide the system running time dimension into a number of time slots, and use  $T = \{0, 1, 2 \dots\}$  to represent the set of time slots for network operation, where the time length of each time slot  $t$  is defined as  $\tau$ . It is assumed that most of the computing tasks of user can be processed and completed in one time slot. Due to the large amount of data, some computing tasks are divided into subtasks for processing [20]. Considering the randomness of task arrival, a two-level queue model is designed to describe the state of computing tasks, namely the user task queue model and MEC server task queue model.

*2.2. Task Generation Model.* In MEC model, it is assumed that the time interval for mobile users to generate tasks obeys Poisson distribution, and user  $n$  generates  $k_n$  mutually independent tasks, which are defined as  $K_n = \{1, 2, \dots, k_n\}$ . The attribute of task  $i$  is defined as  $\varphi_i = \{id_n, id_i, sub_i, d_i, c_i, l_i, mem_i, cpu_i\}$ , where  $id_u$  represents the identity (id) of user  $n$  who generated task  $i$ ,  $id_i$  represents the id of task, and  $sub_i$  represents the time when the user submits the task.  $d_i$  (bits as a unit) represents the amount of task data,  $c_i$  (CPU revolutions/bit as a unit) represents the number of CPU revolutions required to calculate one bit of task data, and  $l_i = d_i \cdot c_i \cdot mem_i$  and  $cpu_i$ , respectively represent the memory and CPU resources required by computing tasks. Users are mobile and may be located near different base stations at different points in time. Thus, tasks generated by same user may be offloaded to servers in different base stations for processing.

### 2.3. Calculation Model

*2.3.1. Local Calculation Model.* Mobile users themselves have certain computing capabilities. If the user has sufficient computing resources, then tasks can be processed locally. The computing power of local device  $n$  is represented by CPU frequency, which is defined as  $f_{n,l}$ . The processing time of task on local calculation model only considers the calculation time. Therefore, the local processing time of task  $i$  generated by user  $n$  is defined as

$$t_{i,n} = \frac{l_i}{f_{n,l}}. \quad (1)$$

The power and energy consumption of task  $i$  processed locally by user  $n$  are, respectively, defined as

$$\begin{cases} P_{i,n} = \gamma(f_{n,l})^3 \\ E_{i,n} = \gamma(f_{n,l})^2 l_i \end{cases}, \quad (2)$$

where  $\gamma$  is the effective switch capacitance.

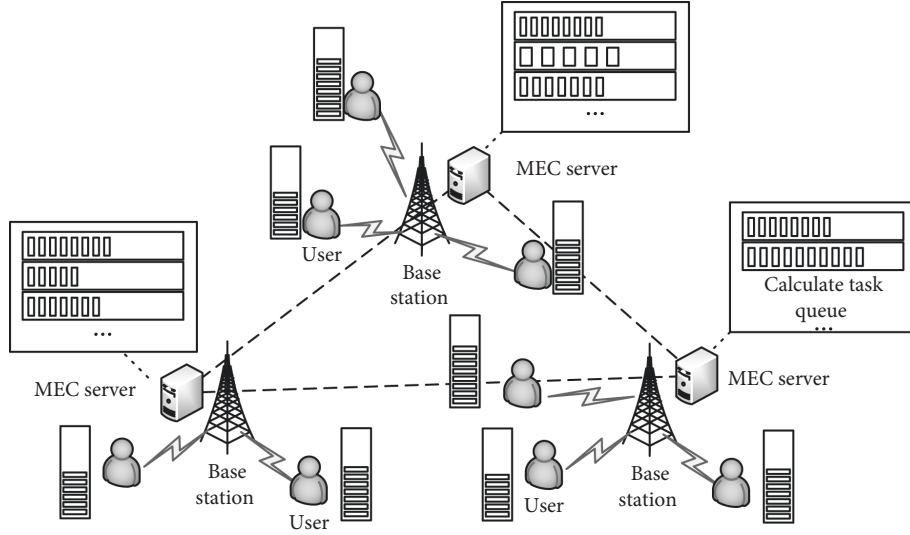


FIGURE 1: System scenario.

**2.3.2. Edge Computing Model.** Due to the insufficient computing resources of local devices, a large number of tasks generated by users cannot be processed on local computing model, but some tasks need to be offloaded to edge computing model for processing [21]. When the task is executed on MEC, the transmission time and calculation time need to be considered, and the amount of data returned by the task is very small, so the transmission time does not consider the time-consuming of result return. Before calculating the transmission time, first define the transmission rate from user device  $n$  to base station  $m$  as

$$v_{n,m} = B \log_2 \left( 1 + \frac{p_n h_{n,m}}{\delta_0 B} \right), \quad (3)$$

where  $B$  is the communication bandwidth;  $p_n$  is the transmission power of user  $n$ .  $\delta_0$  is the noise power spectral density of base station  $m$ ;  $h_{n,m}$  is the channel gain between user  $n$  and base station  $m$ .

The time for task  $i$  generated by user device  $n$  to be offloaded to server  $j$  of base station  $m$  for processing is defined as

$$t_{i,off} = \frac{d_i}{v_{n,m}} + \frac{l_i}{f_{m_j}}, \quad (4)$$

where  $f_{m_j}$  is the CPU frequency of server  $j$  on base station  $m$ .

Same as time-consuming calculation, the energy consumption of task  $i$  generated by user  $n$  and offloaded to server  $j$  of base station  $m$  for processing is defined as

$$E_{i,off} = p_n \frac{d_i}{v_{n,m}} + d_i e_{m_j}, \quad (5)$$

where  $e_{m_j}$  is the energy consumption required to calculate one bit of data.

**2.4. Optimization Goal.** The optimization goal is to reduce the average response time and total energy consumption of

tasks in MEC environment, improve user service quality and save system energy cost [22]. The execution of tasks on computing nodes will be constrained by some network hardware environments [23]. Suppose that the maximum number of tasks that can be executed in parallel on the computing node is  $\Gamma$ , if the number of tasks is less than  $\Gamma$ , new tasks can be received; Otherwise, you need to wait for the resource release task to be executed. In addition, a new task can be processed only when the network resource required by the executing task and the new task is less than the total resources. The mathematical expression of the objective optimization problem is as follows:

$$\begin{cases} \min_{\{q\}} \bar{T} \text{ and } E_{total} \bar{T} = \sum_{k=1}^K \frac{\hat{T}_i}{k}, \hat{T}_i = T_i^{\infty} - sub_i \\ E_{total} = \sum_{i=1}^{k_n} E_i \end{cases}, \quad (6)$$

s.t.  $q \leq \Gamma,$

$$\sum_{i=1}^q mem_i \leq C_1,$$

$$\sum_{i=1}^q cpu_i \leq C_2,$$

where  $q$  is the number of simultaneous tasks, and  $C_1$  and  $C_2$  are memory and CPU capacity respectively, and  $T_i^{\infty}$  is the completion time of task  $i$ ,  $E_i = \begin{cases} E_{i,n}, & \text{local computing} \\ E_{i,off}, & \text{of floa di ng computing} \end{cases}$ .

### 3. Solutions Based on Deep Reinforcement Learning

**3.1. Markov Decision Process Modeling.** The Markov decision process is described by quadruple  $\langle S, A, \psi, R \rangle$ :

- (1)  $S$  is the system state collection. For an incomplete observation system, the set used by edge server to describe the system state only includes the basic information of edge server:  $S = \{S_1, S_2, \dots, S_j\}$ . Among them, let  $S_j$  be a 5-tuple.
- (2)  $a_n^t \in A$  is a finite set of actions, that is, the action of calculating offloading. The set includes the user who decides to uninstall at time  $t$ , and the user's action at time  $t$  is recorded as  $a_n^t$ . When  $a_n^t = 0$ , user  $n$  executes locally. When  $a_n^t = 1$ , user  $n$  offloads the task to the MEC.
- (3)  $\psi$  is the state transition matrix, and  $\psi$  corresponds to the mapping of  $S \times A \times S \rightarrow [0, 1]$ . That is, the probability of transitioning to next state after the end of state  $S$ , after the execution of action  $A$ .
- (4)  $R$  is the reward function. When the user needs to uninstall, the uninstall action will get a positive reward. When the decision makes system overload, a negative reward, or penalty, is given.

Reinforcement learning obtains rewards through reward function  $r_t$  at time  $t$ . For some observable system environments, the remote server can only obtain information about tasks that have been offloaded to the remote [24,25]. Therefore, it is considered that the amount of calculation saved is regarded as a reward for an offloading behavior. In order to better master the use of system resources, a punitive reward will be set. The punitive reward is set to the negative value of absolute value of current system reward, which ensures that the punitive reward value is always negative [26,27]. The punitive reward is expressed as

$$\bar{r}_t = -|r_{t-1}| \quad (7)$$

Markov process corresponds to a sequence of system state transitions, that is, a trajectory sequence  $\Xi = \langle s^0, a^0, s^1, a^1, \dots \rangle$  containing states and actions can be obtained. Strategy  $\pi$  corresponds to the mapping of  $S \times A \rightarrow [0, 1]$ . Deep reinforcement learning maximizes the cumulative reward expectation of  $\Xi$  during the training process to find optimal strategy  $\pi$ .

**3.2. DQN-Based Offload Strategy.** The training process based on DQN offloading strategy is shown in Figure 2.

According to the above figure, the pseudo code of algorithm based on DQN offloading strategy is shown in Algorithm 1.

Based on DQN algorithm, two neural network structures, the current Q-value network and target Q-value network are used. The two have the same neural network structure, but the parameters of their respective structures are different. The definition  $\theta$  represents the parameters of current Q-value network, and  $\theta'$  represents the parameters of target Q-value network. DQN algorithm fits the action value function  $Q(s_t, a_t; \theta)$  through Q-value network with parameter  $\theta$ , which is calculated as follows:

$$Q(s_t, a_t; \theta) = E \left[ \sum_t \chi_t R(s_t, a_t) | s, a \right], \quad (8)$$

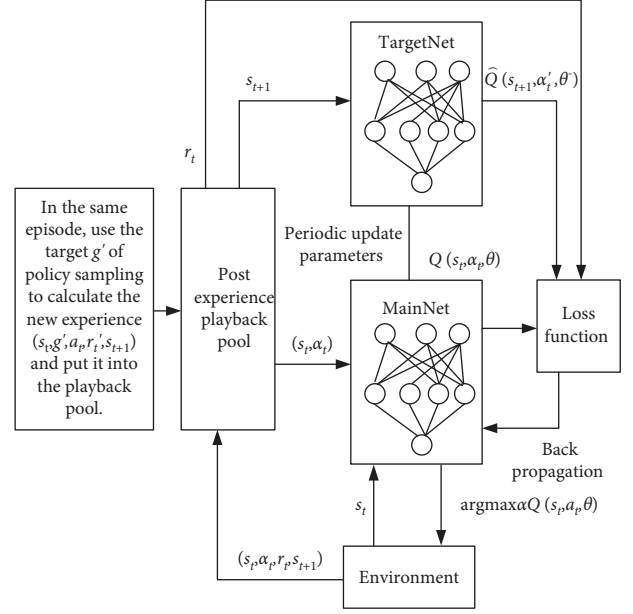


FIGURE 2: Training process offloading strategy based on DQN.

where  $\chi \in [0, 1]$  is the reward discount factor.

Then select the optimal action based on value of each action generated by Q-value network:

$$a_t = \arg \max_a Q(s, a; \theta). \quad (9)$$

In order to avoid not selecting the optimal local optimal solution when selecting an action,  $\epsilon$ -greedy strategy is used to select an action. That is, an action is randomly selected with a small probability of  $\epsilon$ , and the optimal action is selected according to (10) with a probability of  $1 - \epsilon$ , so as to obtain the reward value  $r_t$  and next state  $s_{t+1}$ . Then put quadruple  $(s_t, a_t, r_t, s_{t+1})$  into the experience replay library, and sample a batch of  $(s_t, a_t, r_t, s_{t+1})$  into the neural network for training. When action  $a_t$  is executed, the Q-value corresponding to action space  $a_t$  is updated, according to Bellman formula:

$$Q(s, a; \theta) = R(s, a) + \chi \max_{a'} Q(s_{t+1}, a'; \theta'). \quad (10)$$

Then minimize Loss function to update the parameters of current Q-value network. Loss function represents the predicted value of square error loss between the current Q-value network and target Q-value network. The smaller the value, the better the neural network is optimized. Generally expressed as

$$L(\theta) \approx E \left[ \left( r_t + \chi \max_{a'} Q(s_{t+1}, a'; \theta') - Q(s_t, a_t; \theta) \right)^2 \right]. \quad (11)$$

Then the target Q-value network is updated with a delay.

## 4. Experiment and Analysis

The platform used in the experiment is Python 3.6, and Tensorflow GPU 1.14 is used for in-depth learning and

```

Input: Target resampling strategy  $S$ , Reward function  $R: S \times A \times g \rightarrow R$ 
Begin
(1) Initialize replay pool
(2) For episode = 0, 1, 2, ...,  $m$ 
    do Initialize a state  $s_0$  and a target  $g$ ;
(3) For  $t=0, 1, 2, \dots, T-1$ 
    do Use behavior strategies to select actions  $a_t$ 
    Execute action  $a_t$  and observe the new state  $s_{t+1}$ 
(4) End for
(5) For  $t=0, 1, 2, \dots, T-1$ 
    do Calculate immediate rewards
    Put  $(s_t, g, a_t, r_t, s_t)$  this experience is stored in the playback pool
    Resample a batch of target  $G$  using the target resampling policy  $S$ 
(6) For  $g' \in G$ 
    do Calculate new immediate rewards  $r'$ 
    Put  $(s_t, g', a_t, r', s_{t+1})$  this new experience is stored in the playback pool
(7) End for
(8) End for
(9) For  $t=0, 1, 2, \dots, N$ 
    do Sample some minibatch from the replay pool
    Calculate the loss function and update the network parameters
(10) End for
(11) End for
End

```

ALGORITHM 1: Pseudo code of offloading strategy based on DQN.

optimization training. Meanwhile, the simulation parameter settings are shown in Table 1.

In addition, the proposed strategy is compared with reference [13], reference [18], and reference [19] to demonstrate its advantages. Among them, reference [13] proposed a Multi-Agent Reinforcement Learning Algorithm for computing offload of Internet of things edge computing network; Reference [18] formulated a resource allocation strategy based on orthogonal and non-orthogonal multiple access schemes; Reference [19] uses Pareto archive evolution strategy to achieve multi-objective resource allocation.

**4.1. Analysis of Energy Consumption Results.** The relationship between the number of users and energy consumption for the four strategies is shown in Figure 3.

It can be seen from Figure 3 that the energy consumption of each strategy basically shows an upward trend. However, the rise of proposed strategy has slowed down. When the number of users is 110, the final energy consumption is about 2500J. This is because too many users lead to full load of edge computing nodes, so tasks are offloaded to higher-performance cloud data centers, keeping the energy consumption of proposed strategy to a low level. Besides, it comprehensively considers local and offloading energy consumption using DQN to obtain the optimal offloading plan, which can effectively reduce energy consumption. In reference [13], although multi-agent reinforcement learning algorithm is used to obtain the optimal offloading plan, the cloud data center is not considered, so the energy consumption is increasing rapidly. The other two strategies are difficult to handle increased number of users, and the energy consumption is higher, exceeding 3500J.

TABLE 1: Experimental parameter and hyperparameter setting.

Parameter	Value
$c_i$ (Cycles/bit)	500
$f_{n,i}$ (GHz)	Unif (0.5,1.0)
$f_{m,j}$ (GHz)	Unif (5.0,10.0)
$p_n$ (mW)	150
$C$ (%)	150
$D_{n,m}$ (m)	randint (50,200)
$C_1$ (GB)	32
$\delta_0$ (dBm/Hz)	-175
$\Omega$	12
$\chi$	0.7

**4.2. Analysis of Time Delay Results.** Similarly, the relationship between users and time delay under the four strategies is shown in Figure 4.

It can be seen from Figure 4 that reference [18] preferentially chooses to execute tasks locally to meet the requirements of delay-sensitive tasks. If the computing resources are insufficient, it will turn to high-level devices for offloading, so the delay is almost the lowest, no more than 5s. However, the strategy in reference [19] tends to preferentially offload tasks to edge nodes, and the increase in the number of users will reasonably uninstall some tasks, because the computing resources of edge nodes are in short supply and need to be queued for use, the delay will increase suddenly. As the number of users further increases, tasks are reasonably offloaded, which can alleviate time delay to a certain extent. But because of transmission link, although there is no need to queue up, a lot of time is lost in the transmission process. Even if the task continues to increase, time delay will stabilize in a higher range, about 17s.

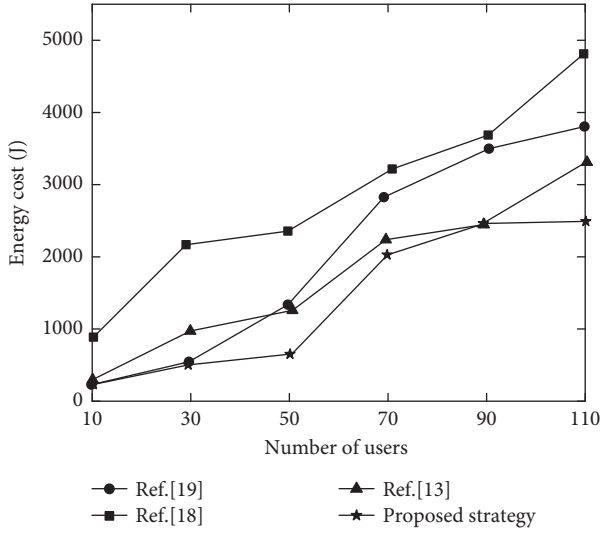


FIGURE 3: Comparison of energy consumption of different strategies.

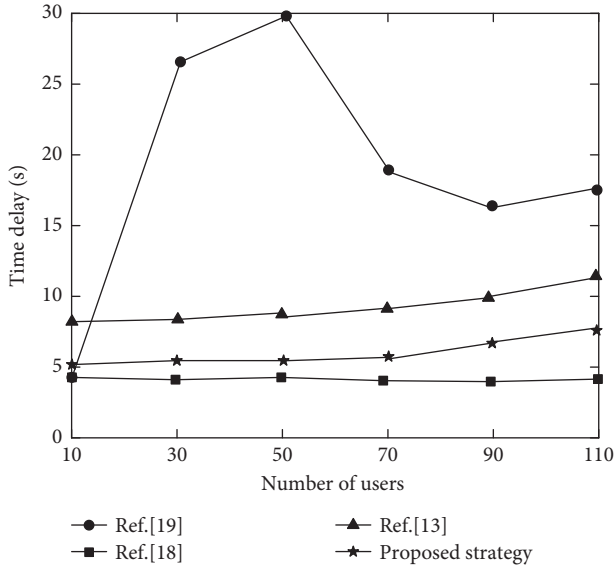


FIGURE 4: Comparison of time delay of different strategies.

However, the delays of reference [13] and proposed strategy are relatively stable. The proposed strategy fully considers the time and energy consumption of local and offloading to MEC execution, and solving offloading scheme by reinforcement learning can greatly reduce delay.

**4.3. Analysis of Load Balancing Rate Results.** Figure 5 shows the relationship between users and load balancing ratios under the four strategies.

It can be seen from Figure 5 that the overall load balancing ratio of reference [18] strategy is relatively high. This is because it focuses on local execution, and task offloading starts from the device with the lowest performance, so as long as the device performing the task is almost fully loaded. Although some pressure was shared between 30 and 70 by offloading to edge nodes, the resources of edge nodes were

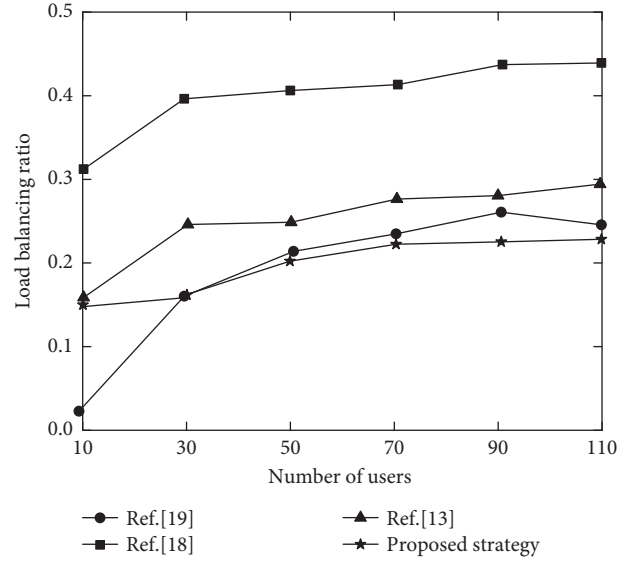


FIGURE 5: Load balancing rate of different strategies.

quickly occupied. However, the strategy in reference [19] tends to be offloaded to MEC server, so the load balancing rate is low. This can maintain a high utilization rate for a relatively large number of edge node clusters with moderate performance. Reference [13] used multi-agent reinforcement learning algorithms for task offloading, but the load balancing rate is low. However, the algorithm performance still needs to be improved compared with DQN, so the load balancing rate of proposed strategy is the lowest, about 0.23. Reasonable utilization of users, MEC and cloud center can greatly reduce the load balancing rate.

**4.4. Impact of Different Similarity Measurement Methods on Algorithm Execution Efficiency.** According to the pipeline model, bandwidth resource bottleneck is the first dilemma faced in the offloading process. Ensuring the effective use of bandwidth resources, rather than blindly offloading too many tasks, is the key to rational use of system resources. With the increase in the number of users, the four strategies are shown in Figure 6 for network and server usage.

It can be seen from Figure 6 that compared with other strategies, the broadband utilization rate and computing resource utilization rate of proposed strategy is relatively low. Among them, the broadband utilization rate is always between 0.1 and 0.3, and computing resource utilization rate is roughly between 0.2 and 0.45. Since the proposed strategy always occupies a lower bandwidth in the decision-making process, DQN strategy is used to reasonably offload computing tasks, thereby avoiding bottlenecks in network transmission. At the same time, because fewer broadband resources are occupied, higher revenue can be obtained for servers. Reference [13] performed computational offloading based on multi-agent reinforcement learning algorithm. Although the task offloading can be completed well, MEC server has a higher requirement for computing power, so it occupies more computing resources. Reference [18] and reference [19] lacked high-performance processing algorithms and cannot balance

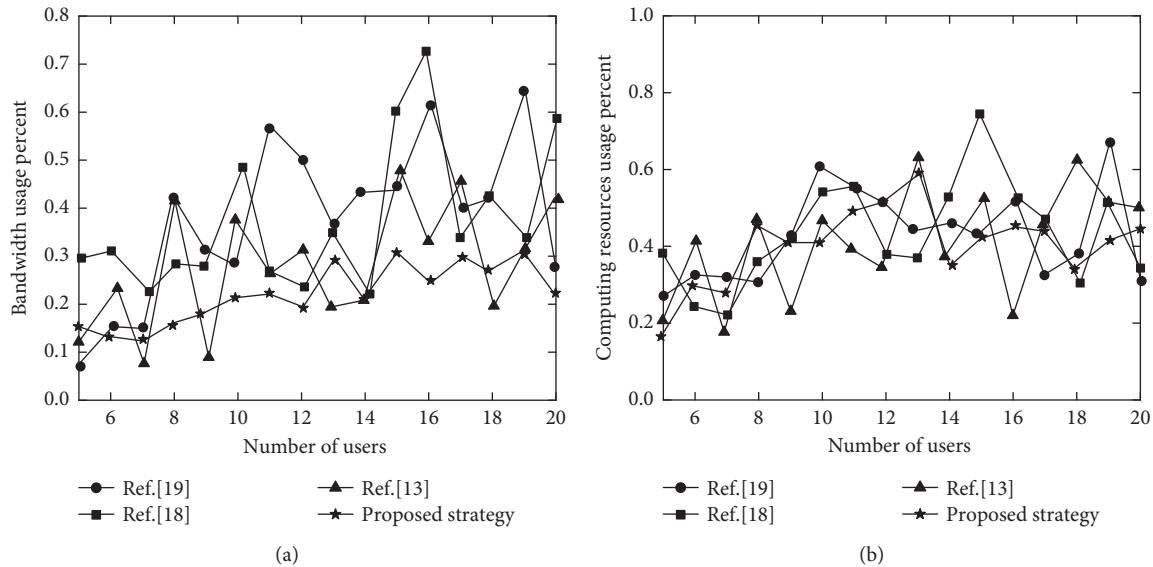


FIGURE 6: Changes of network and server load with the number of users. (a) Bandwidth usage percent. (b) Computing resources usage percent.

task offloading. Thus, the broadband utilization rate and computing resource utilization rate fluctuate greatly and are at a high value.

## 5. Conclusion

With the rapid development of IoT and 5G technology, a series of new applications with computationally intensive and delay-sensitive features such as virtual reality, augmented reality and face recognition continue to emerge. In order to solve the problem of insufficient local computing power, the proposed strategy offloads some tasks to the edge of network, and builds a mobile edge system model with multi-MEC server and multi-user. This model improves the task processing capability of system by solving goal of minimizing. Besides, DQN strategy is used to obtain an offloading plan that minimizes the average response time of system tasks and total energy consumption, so as to allocate computing resources reasonably.

The proposed strategy has certain value and significance for theoretical research and practical application. However, due to resource constraints such as mobile devices, servers and base stations, experiments can only be carried out in a simulated environment that is as close to the actual situation as possible. In the future research work, we will further consider conducting physical experiments in a real environment to provide solutions to practical problems.

## Data Availability

The data used to support the findings of this study are included within the study.

## Conflicts of Interest

The authors declare that there are conflicts of interest regarding the publication of this study.

## Acknowledgments

This work was supported by The 2019 Anhui Province University Outstanding Top Talent Cultivation Funding Project, "Application Research of Task Scheduling Based on Energy Awareness in Heterogeneous Hybrid Cloud Environment" (Item Number: gxgnfx2019050) and The Second Batch of Industry-University Cooperation Collaborative Education Projects in 2020, Blockchain + big data professional teacher training and course construction for new engineering (Item Number: 202101207023), Demonstration Project of Provincial Grassroots Teaching and Research Office of Software Engineering: (Item Number: 2020JSJFJXZZ417).

## References

- [1] W. Wen, Y. Cui, T. Q. S. Quek, F. C. Zheng, and S. Jin, "Joint optimal software caching, computation offloading and communications resource allocation for mobile edge computing," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 7, pp. 7879–7894, 2020.
- [2] T. Alfakih, M. M. Hassan, A. Gumaie, C. Savaglio, and G. Fortino, "Task offloading and resource allocation for mobile edge computing by deep reinforcement learning based on SARSA," *IEEE Access*, vol. 8, no. 6, pp. 54074–54084, 2020.
- [3] H. Zhang, Z. Wang, and K. Liu, "V2X offloading and resource allocation in SDN-assisted MEC-based vehicular networks," *China Communications*, vol. 17, no. 5, pp. 266–283, 2020.
- [4] W. Wu, F. Zhou, R. Q. Hu, and B. Wang, "Energy-efficient resource allocation for secure NOMA-enabled mobile edge computing networks," *IEEE Transactions on Communications*, vol. 68, no. 1, pp. 493–505, 2020.
- [5] X. Sun, J. Zhao, X. Ma, and Q. Li, "Enhancing the user experience in vehicular edge computing networks: an adaptive resource allocation approach," *IEEE Access*, vol. 7, no. 7, pp. 161074–161087, 2019.



- [6] X. Zhang, X. Zhu, M. Chikuvanyanga, and M. Chen, "Resource sharing of mobile edge computing networks based on auction game and blockchain[J]," *EURASIP Journal on Applied Signal Processing*, vol. 2, no. 1, pp. 1–23, 2021.
- [7] K. Zhang, Y. Mao, S. Leng, S. Maharjan, A. Vinel, and Y. Zhang, "Contract-theoretic approach for delay constrained offloading in vehicular edge computing networks," *Mobile Networks and Applications*, vol. 24, no. 3, pp. 1003–1014, 2019.
- [8] G. Li and X. Song, "Data distribution optimization strategy in wireless sensor networks with edge computing," *IEEE Access*, vol. 8, no. 5, pp. 214332–214345, 2020.
- [9] C. Yang, Y. Liu, X. Chen, W. Zhong, and S. Xie, "Efficient mobility-aware task offloading for vehicular edge computing networks," *IEEE Access*, vol. 7, no. 8, pp. 26652–26664, 2019.
- [10] S. Min, W. Yanting, W. Xijun, and L. Jiandong, "Energy-efficient multiuser partial computation offloading with collaboration of terminals, radio access network, and edge server [J]," *IEEE Transactions on Communications*, vol. 68, no. 3, pp. 1524–1537, 2019.
- [11] C. Li, W. Chen, J. Tang, and Y. Luo, "Radio and computing resource allocation with energy harvesting devices in mobile edge computing environment," *Computer Communications*, vol. 145, no. 9, pp. 193–202, 2019.
- [12] Y. Liao, L. Shou, Q. Yu, Q. Ai, and Q. Liu, "Joint offloading decision and resource allocation for mobile edge computing enabled networks," *Computer Communications*, vol. 154, no. 8, pp. 361–369, 2020.
- [13] X. Liu, J. Yu, Z. Feng, and Y. Gao, "Multi-agent reinforcement learning for resource allocation in IoT networks with edge computing," *China Communications*, vol. 17, no. 9, pp. 220–236, 2020.
- [14] F. Guo, F. R. Yu, H. Zhang, H. Ji, M. Liu, and V. C. M. Leung, "Adaptive resource allocation in future wireless networks with blockchain and mobile edge computing," *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 1689–1703, 2020.
- [15] X. Liu, J. Yu, J. Wang, and Y. Gao, "Resource allocation with edge computing in IoT networks via machine learning," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3415–3426, 2020.
- [16] X. Chen, Z. Liu, Y. Chen, and Z. Li, "Mobile edge computing based task offloading and resource allocation in 5G ultra-dense networks," *IEEE Access*, vol. 7, pp. 184172–184182, 2019.
- [17] Q. Zhang, L. Gui, F. Hou, J. Chen, S. Zhu, and F. Tian, "Dynamic task offloading and resource allocation for mobile-edge computing in dense cloud RAN," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3282–3299, 2020.
- [18] M. Zeng and V. Fodor, "Energy minimization for delay constrained mobile edge computing with orthogonal and non-orthogonal multiple access," *Ad Hoc Networks*, vol. 98, pp. 102060–102060.13, 2020.
- [19] Q. Liu, R. Mo, and X. Xu, "Multi-objective resource allocation in mobile edge computing using PAES for Internet of Things," *Wireless Networks*, vol. 5, no. 3, pp. 1–13, 2020.
- [20] T. Li, K. Wang, K. Xu, K. Yang, C. S. Magurawalage, and H. Wang, "Communication and computation cooperation in cloud radio access network with mobile edge computing," *CCF Transactions on Networking*, vol. 2, no. 1, pp. 43–56, 2019.
- [21] Z. Yang, C. Pan, J. Hou, and M. Shikh-Bahaei, "Efficient resource allocation for mobile-edge computing networks with NOMA: completion time and energy minimization," *IEEE Transactions on Communications*, vol. 67, no. 11, pp. 7771–7784, 2019.
- [22] H. Wang, H. Ke, G. Liu, and W. Sun, "Computation migration and resource allocation in heterogeneous vehicular networks: a deep reinforcement learning approach," *IEEE Access*, vol. 8, no. 8, pp. 171140–171153, 2020.
- [23] J. Du, F. R. Yu, X. Chu, J. Feng, and G. Lu, "Computation offloading and resource allocation in vehicular networks based on dual-side cost minimization," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1079–1092, 2019.
- [24] H. Tang, C. Li, J. Bai, J. Tang, and Y. Luo, "Dynamic resource allocation strategy for latency-critical and computation-intensive applications in cloud-edge environment," *Computer Communications*, vol. 134, no. 6, pp. 70–82, 2019.
- [25] Q. V. Pham, L. B. Le, S. H. Chung, and W. J. Hwang, "Mobile edge computing with wireless backhaul: joint task offloading and resource allocation," *IEEE Access*, vol. 7, no. 99, pp. 16444–16459, 2019.
- [26] H. Guo, J. Zhang, J. Liu, and H. Zhang, "Energy-aware computation offloading and transmit power allocation in ultradense IoT networks," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4317–4329, 2019.
- [27] M. Yan, C. A. Chan, W. Li, L. Lei, A. F. Gygax, and C. L. I, "Assessing the energy consumption of proactive mobile edge caching in wireless networks," *IEEE Access*, vol. 7, no. 99, pp. 104394–104404, 2019.

## Research Article

# Image Semantic Segmentation Method Based on Deep Fusion Network and Conditional Random Field

Shuo Wang <sup>1</sup> and Yi Yang <sup>2</sup>

<sup>1</sup>*School of Energy and Intelligence Engineering, Henan University of Animal Husbandry and Economy, Zhengzhou, Henan 450044, China*

<sup>2</sup>*School of Information Engineering, Henan University of Animal Husbandry and Economy, Zhengzhou, Henan 450000, China*

Correspondence should be addressed to Yi Yang; [yangyi2017@bupt.edu.cn](mailto:yangyi2017@bupt.edu.cn)

Received 20 January 2022; Revised 23 March 2022; Accepted 27 April 2022; Published 14 May 2022

Academic Editor: Laith Abualigah

Copyright © 2022 Shuo Wang and Yi Yang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Aiming at the problems of missing points and wrong points in image semantic segmentation under complex background and small target, an image semantic segmentation method based on the fully convolution neural network and conditional random field is proposed. First, the deconvolution fusion structure is added to the fully convolution neural network to build a deep fusion network. The multiscale features are automatically obtained through the deep fusion network, and the shallow detail information and deep semantic information are fused to improve the processing accuracy of image rough segmentation. Then, the bivariate potential function of the conditional random field is optimized based on the convolution neural network, and it is used for image fine segmentation to obtain the final image segmentation result. Finally, the proposed method is experimentally analyzed based on the Cityscapes dataset. The results show that the proposed method can achieve accurate image segmentation, and the area under the segmentation curve of the overall size target is 93.6%, which is better than other methods.

## 1. Introduction

As an important research direction in the field of computer vision, the ultimate goal of image semantic segmentation is to use the computer to simulate human vision to quickly perceive objects and analyze environmental information, realize image semantic understanding and reasoning, and make corresponding action feedback according to the high-resolution information input by vision [1, 2]. Specifically, image semantic segmentation refers to identifying the specific target and location in the image from the pixel level and depicting the scene contour with different colors through target detection.

At present, semantic segmentation has been widely used in medical images, remote sensing systems, clothing, security, and transportation [3]. In medical image analysis, semantic segmentation is applied to tumor image analysis and tooth lesion judgment [4]. In the satellite remote sensing system, it is used to segment, locate, and label the geographical objects such as roads,

rivers, forests, crops, and buildings in the remote sensing image, which can save a lot of manual labeling cost and time cost [5]. In the field of unmanned driving, the vision-based processing method is still an important part, and image semantic segmentation is regarded as an important technology supporting automatic driving application scenarios [6, 7].

Although image semantic segmentation has been widely used, there are still some bottlenecks that limit its development. For example, different kinds of objects with similar appearances are difficult to distinguish, and smaller objects are easy to lose details and specific contours, which are all problems to be solved in existing image semantic segmentation [8]. In order to solve these difficulties, most early image semantic segmentation technologies are based on traditional methods, mainly including segmentation methods based on the threshold, edge detection, and region [9]. With the emergence of deep learning, the image semantic segmentation method based on deep learning gradually replaces the traditional methods, and its accuracy,

speed, and other performance indicators have been greatly improved [10, 11].

At present, there has been some research on image semantic segmentation in various fields at home and abroad. The traditional image segmentation methods pay more attention to the separation of target and background in the image. There are six traditional image segmentation methods based on the threshold, edge detection, graph theory, region, clustering, and specific theoretical tools. For example, reference [12] analyzed the principles, advantages, and disadvantages of image semantic segmentation based on traditional methods and deep learning methods and pointed out that deep learning network had better optimization results than traditional methods. Reference [13] proposed a new image redirection method using semantic segmentation and pixel fusion, which could finely reassign the scaling factor for each region according to the semantic segmentation results, so as to effectively reduce the geometric distortion in the process of image redirection, but the detection efficiency needs to be improved. Reference [14] proposed an encoder-decoder architecture, which used global and local semantics to solve the problem of automatic image coloring and fine-tune the low-level coding features through scene context classification to integrate the global image style, but it is easy to lose specific details for the contour of small objects. Reference [15] proposed a weak supervision framework for zero sample semantic segmentation, which could segment images with target categories without any pixel-level marker instances, but the effect of image semantic segmentation in a complex environment was poor.

Now, the deep learning algorithm has been widely used in image semantic segmentation [16, 17]. Reference [18] proposed an image semantic segmentation method based on deep learning, which was different from the traditional image segmentation and improved the robustness, timeliness, and accuracy of lane semantic segmentation, but the analysis efficiency and sensitivity of randomly changing images need to be improved. Reference [19] proposed an image semantic segmentation algorithm based on the fully convolution neural network. Reference [20] proposed a crack detection method based on deep learning semantic segmentation. Through the photos of a large number of concrete structures with adverse conditions such as shadow and dirt, the accuracy of the developed method was studied. It was found that not only the crack area could be detected but also the traces of tie rod hole and formwork could be removed, so as to improve the detection accuracy.

Based on the above analysis, aiming at the problem of image semantic segmentation with random information in a complex environment, an image semantic segmentation method based on the convolution neural network and conditional random field is proposed. In order to improve the performance of image semantic segmentation, the deconvolution structure is fused into the fully convolution neural network to obtain multiscale features. The shallow detail information and deep semantic information are combined for image rough segmentation, which effectively improves the processing accuracy. The proposed method

adopts a fully connected conditional random field model, which can make better use of spatial context information to realize boundary location. The edge contour of the segmented image is clear and close to the label image.

## 2. Image Semantic Segmentation Based on Deep Fusion Network Combined with Conditional Random Field

*2.1. Algorithm Framework.* Before the era of deep learning (DL), the traditional semantic segmentation focuses on the low-order visual information of image pixels [21]. Due to the lack of algorithm training in traditional methods, the algorithm has high complexity, slow convergence, and long time-consuming, and the segmentation effect is often not optimistic. There is no major breakthrough in a long time. With the rapid development of deep learning, breakthroughs have been made in traditional tasks such as image classification. After computer vision entered the era of deep learning, semantic segmentation also begins to try to use the method of deep learning. Semantic segmentation methods based on deep learning continue to emerge, which repeatedly refresh the accuracy of image semantic segmentation.

Aiming at the defects of a large receptive field and weak edge based on a fully convolution network, the proposed method first uses a fully convolution neural network for rough segmentation and then uses a conditional random field for fine segmentation (enhancing edge constraint). Since deep learning, especially convolution neural network, began to be applied to image semantic segmentation, such a general framework has gradually formed, as shown in Figure 1.

The front-end image semantic rough segmentation of the network framework is mainly based on the fully convolution network model. The back-end image fine segmentation mostly adopts the undirected probability graph model such as conditional random field, which uses the univariate potential function to describe the information of the current pixel and uses the bivariate potential function to express the information between two pixels. The front-end module is essentially the same as the feature extraction in the traditional method. The back-end module uses the conditional random field to explain the relationship between the essences of transactions and makes pixel-level prediction.

### 2.2. Image Semantic Rough Segmentation Based on Fully Convolution Neural Network

*2.2.1. Fully Convolution Neural Network.* Since Krizhevsky won the championship in a large-scale image recognition competition by using deep convolution network model, convolution neural network technology has been concerned and applied to related fields by researchers at home and abroad. The advantage of a convolution neural network is that the multilayer structure automatically learns different levels of features. The shallow feature map in the network has richer detailed information. The feature map extracted after multiple convolutions, and pooling operations have deeper semantic information. The fully convolution neural network

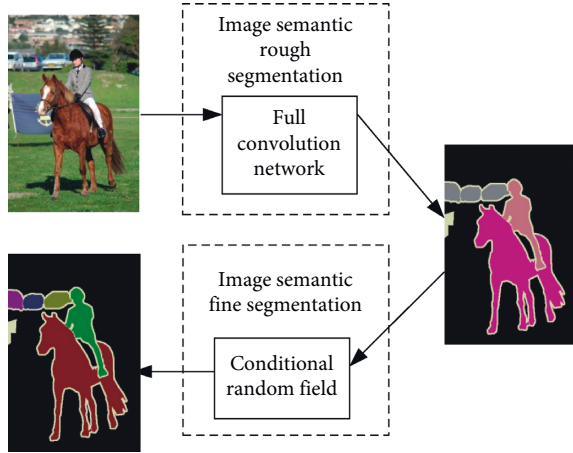


FIGURE 1: Framework of image semantic segmentation network.

is mainly composed of four parts: input layer, convolution layer, pooling layer, and output layer.

- (1) Input layer: it reads data as an array with dimension  $c \times h \times d$ , where  $c$  and  $h$  are sizes of data, and  $d$  is a feature or channel dimension
- (2) Convolution layer: it is also known as the feature extraction layer, which is a process of convolution of input data using filters with specific weights. The features extracted by different convolution kernels are different. The convolution results are output through the activation function. The activation function adopted by the proposed method is rectified linear units (ReLU), which is mathematically expressed as

$$f(x) = \max(0, x). \quad (1)$$

- (3) Pooling layer: it is also known as the down-sampling layer. In order to avoid the problem of fitting too many parameters, the pooling layer is used to reduce the amount of data and speed up the network training on the basis of retaining useful information. Common pooling operations include maximum pooling and average pooling. The convolution layer and pooling layer can be defined as

$$y_{ab} = l_{ks} \left( \left( x_{sa+\delta a, sb+\delta b} \right)_{0 \leq \delta a, \delta b < k} \right), \quad (2)$$

where  $x_{ab}$  is the data vector at the position  $(a, b)$ ,  $y_{ab}$  is the output vector after layer operation  $(a, b)$ ,  $k$  is the convolution kernel size,  $s$  is the step size or down-sampling factor, and  $l_{ks}$  is the type of layer (representing the matrix multiplication or average pooling operation of the convolution layer and the output layer of ReLU function).

- (4) Output layer: it outputs the probability vector of each pixel belonging to each semantic category through the Softmax classifier

**2.2.2. Deconvolution Fusion Structure.** The pooling operation in the fully convolution network model training not only reduces the image size but also loses the rich detail

information in the image, resulting in low semantic segmentation accuracy [22]. Therefore, deconvolution is used to restore the original image size and realize pixel to pixel image semantic segmentation. At the same time, in order to obtain more accurate semantic segmentation results, the fully convolution network adds a fusion structure, and the shallow detail information in the network model is introduced by combining the pooling layer results of different scales and the final convolution layer results.

(1) *Deconvolution operation.* Deconvolution operation is the operation opposite to convolution operation in the forward and backpropagation of neural network model. Taking the deconvolution operation in the Caffe framework as an example, first, the forward and backpropagation process of convolution operation is analyzed. The forward propagation process is as follows:

$$O = \kappa \times F, \quad (3)$$

where  $\kappa$  is the convolution kernel matrix,  $F$  is the image feature matrix, and  $O$  is the output matrix. During backpropagation, according to the matrix differential formula,

$$\frac{\partial Mx + z}{\partial x} = M^T, \quad (4)$$

where  $M$  represents arbitrary matrix,  $M^T$  is the transpose matrix of  $M$ , and  $z$  is arbitrary constant. It can be deduced as

$$\frac{\partial \text{Loss}}{\partial F} = \frac{\partial \text{Loss}}{\partial O} \cdot \frac{\partial O}{\partial F} = \kappa^T \frac{\partial \text{Loss}}{\partial O}, \quad (5)$$

where Loss is the loss function, and  $\kappa^T$  is the transpose matrix of  $\kappa$ . Therefore, deconvolution is the operation of left multiplication  $\kappa^T$  in forward propagation and left multiplication  $(\kappa^T)^T$  in reverse.

(2) *Fusion structure.* In order to make full use of the detailed information of the image, a fusion structure combining shallow detail information and deep semantic information is adopted. The deep fusion network (DFN) model is shown in Figure 2. The fusion structure is mainly composed of a convolution layer, deconvolution layer, and bonding layer, in which the results from different layers are summed and output. The final convolution layer result of the fully convolution network model only contains the information of the last pooling layer. The model obtained by directly deconvoluting this result is called DFN-5. According to the different depths of the combined pooling layer, it is recorded as models DFN-4, DFN-3, DFN-2, and DFN-1, which respectively represent the depth of the combined pooling layer to pool4, pool3, pool2, and pool1.

Taking DFN-4 as an example, the fusion structure deconvolutes the result of the last convolution layer to the output size of pool4 layer and outputs the result after convolution with pool4 layer through the combination layer. Then, it deconvolutes the result of the combination layer to the input image size and obtains the final semantic segmentation result by using the Softmax classifier. The model DFN-3 deconvolutes the results of the combination layer in

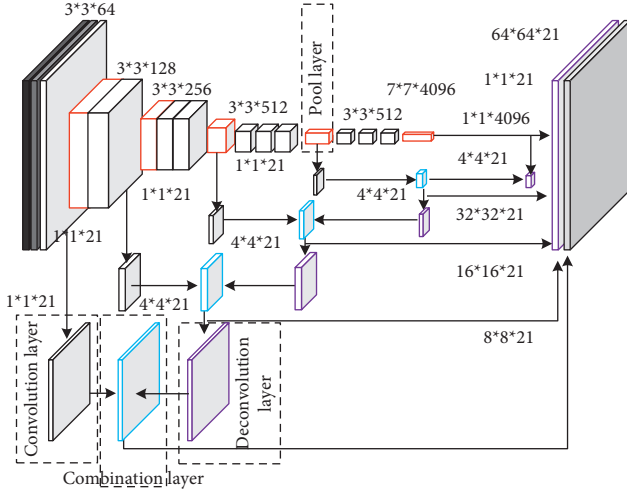


FIGURE 2: Deep fusion network model.

DFN-4 to the output size of the pool3 layer, outputs the results after convolution with the pool3 layer through the combination layer, and then deconvolutes the results of the combination layer to the input image size for Softmax classification. Models DFN-2 and DFN-1 follow the same process.

In the adopted DFN model, the pooling layers are pool1, pool2, pool3, pool4, and pool5 from left to right. The step size of all convolution layers is 1. The size of the pooling layer kernel is 2, and the step size is 2. The deconvolution step size is half of the deconvolution kernel size. Aiming at the defects of large segmentation receptive field and the weak edge of the fully convolution network, the proposed method introduces context information with the help of the second-order potential function of a conditional random field, fully considers the relationship between pixels, improves the accuracy of semantic segmentation, and refines the edge of semantic segmentation results.

**2.3. Image Semantic Segmentation Based on Conditional Random Field.** A conditional random field is applied to image semantic segmentation. Each image is represented by an undirected graph. Each pixel corresponds to the vertex in the undirected graph, and the connection relationship between pixels corresponds to the connecting lines of the vertex in the undirected graph [23, 24]. In the process of image semantic segmentation, different semantic labels are assigned to each pixel. Two pixels with similar location and color features are more likely to be assigned the same semantic label and less likely to be segmented, which corresponds to the probability model in a conditional random field.

Assuming that in the input image, the semantic label of each pixel is expressed as  $X$ , the predicted value matched with the pixel is expressed as  $Y$ , the pixel corresponds to the vertex in the graph, and the association between pixels corresponds to the edge in the graph, so the image can be processed by conditional random field. In a conditional random field, there are usually two potential functions. The

information of each pixel in the image can be expressed by a univariate term, and the correlation between two pixels in the image can be expressed by a bivariate term. In principle, pixels with similar distances will be assigned the same semantic labels as much as possible, and pixels with obvious differences need to be assigned different labels. The evaluation index of "distance" defined by color difference and spatial relative distance ensures that the image can be accurately cut at the edge with a large gradient to a certain extent [25, 26]. Different from the ordinary conditional random field, the bivariate term in the fully connected conditional random field expresses the correlation between each pixel and all other pixels in the image.

In the proposed method, the image semantic fine segmentation module adopts the fully connected conditional random field model, which is an undirected graph model. The vertices in the graph match the pixels in the image, and the edges in the graph match the association between the pixels in the image. The energy function of the model is as follows:

$$E(Y) = \sum_{Y_i \in \mathcal{V}} \phi(Y_i^u) + \sum_{Y_i, Y_j \in \mathcal{E}} \psi(Y_i^u, Y_j^v), \quad (6)$$

where  $Y_i^u \in \{0, 1\}$  represents whether the pixel  $i$  has a label  $u$ .  $\forall u \in L$  represents a set of a label  $L$ ;  $Y, \mathcal{V}, \mathcal{E}$  represent a set of potential variables, node sets, and edge sets, respectively;  $\phi(Y_i^u)$  is a univariate potential function used to measure the cost of assigning the category label  $u$  to the pixel  $i$ . For example, if the pixel  $i$  belongs to the first category label rather than the second category label,  $\phi(Y_i^1 < Y_i^2)$  can be obtained;  $\psi(Y_i^u, Y_j^v)$  is a bivariate potential function, which is used to measure the penalty of assigning labels  $u$  and  $v$  to pixels  $i$  and  $j$ .

The univariate potential function represents the classification of each pixel, and the bivariate potential function represents a set of smoothing constraints.  $\phi(Y_i^u)$  is usually defined as

$$\phi(Y_i^u) = -\ln P(Y_i^{u=1} = 1|I), \quad (7)$$

where  $P(Y_i^u = 1|I)$  represents the probability that the pixel  $i$  belongs to the category label  $u$ ;  $I$  is the pixel position coordinate. The smooth bivariate term is usually defined as

$$\psi(Y_i^u, Y_j^v) = \lambda(u, v)D(i, j), \quad (8)$$

where  $\lambda(u, v)$  represents the penalty when any pair of labels appear globally at the same time. For example, when the label  $u$  and  $v$  do not exist at the same time, the output value should be very large;  $D(i, j)$  is the distance between pixels, which is defined as  $D(i, j) = \omega_1 \|I_i - I_j\|^2 + \omega_2 \| [X_i Y_i] - [X_j Y_j] \|^2$ , where  $I_i$  represents a feature vector, such as the RGB value extracted from the pixel  $i$ ,  $X_i$  and  $Y_i$  represent the coordinates of the pixel positions,  $\omega_1$  and  $\omega_2$  represent the constant terms. If two pixels are close and look similar, they tend to have consistent labels.

However, the choice of such a bivariate potential function model has the following two defects: (1) although the first term of the bivariate potential function can capture the consistency frequency between two labels in the training

data, it ignores the spatial context between objects. For example, people may appear next to the table, but they will not appear at the bottom of the table. This spatial context relationship is a fusion of patterns; that is, there are different configurations of the positional relationship between objects in different images. (2) It only defines the pairwise relationship between pixels but ignores the high-level semantic interaction between them.

In order to solve these two problems, the proposed method uses the output result based on the fully convolution neural network as the univariate potential function of the fully connected conditional random field, and the original bivariate potential function model is replaced by the following formula:

$$\psi(Y_i^u, Y_j^v) = \sum_{q=1}^Q \gamma_q u_q(i, u, j, v) \sum_{\forall w \in N_j} D(j, w) P_w^v, \quad (9)$$

where  $\sum_{q=1}^Q \gamma_q u_q(i, u, j, v)$  represents the fusion of local label context, as the punishment for giving labels in the local area, in which  $Q$  represents the number of different parts in the fusion;  $\gamma_q$  is an indicator variable, similar to a Boolean value, indicating which part is activated and defined as  $\gamma_q \in \{0, 1\}$ , and  $\sum_{q=1}^Q \gamma_q = 1$ . A more intuitive expression is as follows: pixel  $j$  is the adjacent pixel of pixel  $i$ ; that is,  $j \in N_i$ .  $(i, u)$  represents that the pixel  $i$  is assigned as the label  $u$ .  $\lambda(i, u, j, v)$  is the label cost based on the relative position relationship between  $(i, u)$  and  $(j, v)$ . For example, if two labels represent "person" and "table", and the pixel of "person" is located at the lower part of the pixel of "table", the value of the learned penalty function should be large.  $\sum_{\forall w \in N_j} D(j, w) P_w^v$  basically simulates a penalty function term involving three pixels:  $i$ ,  $j$ , and the adjacent points of  $j$ . If  $(i, u)$  and  $(j, v)$  are consistent,  $(i, u)$  should be consistent with the adjacent pixel  $(w, v)$ ,  $\forall w \in N_j$  of the pixel  $j$ .

**2.4. Auxiliary Loss.** There are many branches in the proposed network, and the learning contents of each branch are different. In order to better supervise the training process of each branch, the auxiliary loss is added to the network during training [27]. In the auxiliary loss, first, through convolution, batch standardization, ReLU, the feature map is obtained, and the number of channels is the same as the number of categories. Unlike the feature fusion module, because it is the final output result, the feature map is upsampled to the input size by bilinear interpolation, and finally, the loss is calculated.

### 3. Experiment and Analysis

In the experiment, Cityscapes dataset is used for analysis and demonstration. Cityscapes is a dataset focusing on the understanding of urban street scenes. Cityscapes contains 30 categories (the actual label range is 0–33, i.e., 34 categories), of which 19 classes are used for segmentation tasks. The data are collected during the day, and the sampling conditions are diverse, covering three different seasons of spring, summer, and autumn and different weather conditions (excluding extreme weather environments). The data contain a large number of

small objects, the scene scale is changeable, and the background is complex. The dataset contains a total of 5000 images with fine annotation information and 20000 images with rough marks, which are not commonly used. In the semantic segmentation task, only 5000 images with fine annotation are generally used, including 2800 images for training, 600 images for validation, and 1600 test images. The test set label is also confidential and can only be tested by submitting the results to the server of the dataset. It is worth noting that the images in the dataset have a high resolution of 2048×1024. The segmentation examples of the Cityscapes dataset are shown in Figure 3.

**3.1. Evaluation Index.** When evaluating the proposed segmentation method, the Precision and Recall of the prediction results are calculated to measure the target classification ability and target detection ability of the model. In addition, by setting different confidence thresholds, the Precision-Recall curve of the model, that is, the P-R curve, is drawn to intuitively show the segmentation effect of the model. When calculating the Precision and Recall indicators of the model, first of all, the test results need to be divided into four categories according to the real label: true positive (TP), true negative (TN), false positive (FP), and false negative (FN).

Precision is obtained by calculating the proportion of correctly predicted samples to all predicted samples in the test results, that is, the proportion of the number of correctly detected samples in the total detected samples, which can reflect the classification ability of the model to the target. The formula of Precision is

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \quad (10)$$

Recall is obtained by calculating the proportion of the number of correctly predicted samples to all the real samples in the test results, that is, the proportion of the number of correctly detected samples to the number of real samples, which can reflect the detection ability of the model to the target. The formula of Recall is

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \quad (11)$$

Because the two indicators of Precision and Recall are contradictory, the fixed intersection over union (IOU) threshold and confidence threshold are used to judge whether the detection result is correct. When the IOU threshold and object confidence threshold are high, the calculated Precision value is high, and the Recall value is low. Therefore, in order to comprehensively compare the network performance, it is compared through the P-R curve. The P-R curve takes Precision as the abscissa and Recall as the ordinate. The larger the area surrounded under the curve, the better the performance of the model.

**3.2. P-R Curve Compared with Other Methods.** In order to compare the detection performance of the proposed method with other mainstream methods on objects with different scales, in the experiment, the object is set with the scale of [0,64] pixels as the small target, the object with the scale of

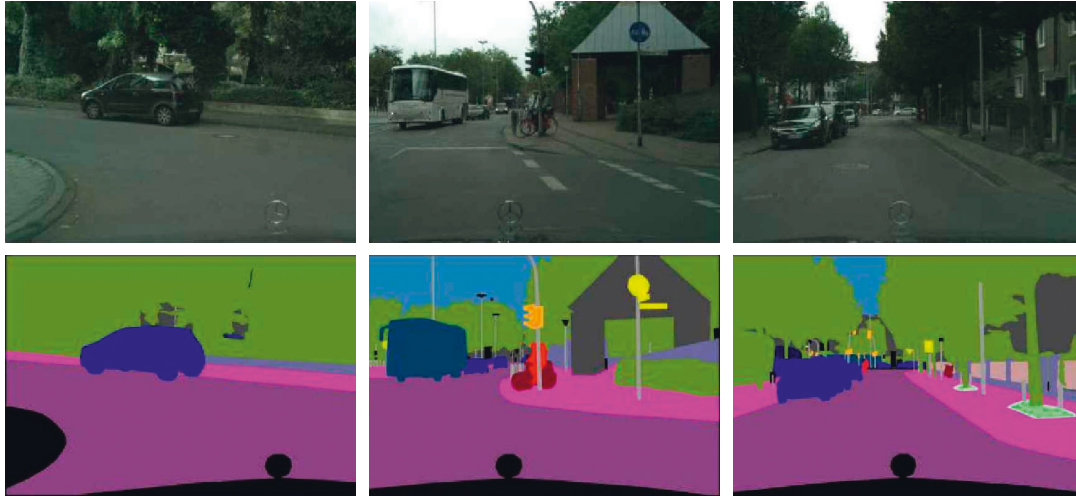


FIGURE 3: Segmentation examples of Cityscapes dataset.

[64,192] pixels as the middle target, and the object with the scale of [192, 500] pixels as the large target. Then, the confidence level is adjusted, and the Precision and Recall values of prediction results of the model in small target, medium target, large target, and the target with overall scale [0, 500] pixels are calculated, respectively. The P-R curve is shown in Figure 4.

It can be seen from Figure 4 that the P-R curve of the proposed method can largely surround the P-R curve of the methods in reference [15] and reference [20] in the detection results of objects with different scales, which shows that the detection of objects with various scales is improved in varying degrees by combining fully convolution neural network and conditional random field. Reference [15] proposes a weak supervision method for zero sample semantic segmentation, which can complete image segmentation without any pixel-level marker instances, but the segmentation effect is poor for most complex environments. Reference [20] proposes a semantic segmentation method based on deep learning, which can realize the image segmentation of small cracks. Therefore, the segmentation effect in [0,64] pixels is not different from that of the proposed method. However, with the increase of pixels, the single segmentation algorithm in reference [20] can not process a large number of images, so the segmentation effect decreases.

**3.3. Comparison of Visual Segmentation Results.** The visual comparison results between the proposed method and the methods in reference [15, 20] on the Cityscapes validation dataset are shown in Figure 5, in which some areas with obvious differences are marked with red dashed boxes.

From the area marked by the red dashed box, it can be seen that the segmentation performance of the proposed method is better than other comparison methods in the details of the Cityscapes validation dataset. First, look at the list of images on the left and their segmentation results. The front wheel of the bike marked with the red dashed box is

partially interspersed with the lamppost. Both the methods in reference [20] and the proposed methods are effective in completely identifying the front wheel of the bike and ignoring the lamppost behind. In the right column of pictures, it can be seen from the red dashed box that the segmentation results of reference [15] mistakenly identify the similar signs on the wall as traffic warning signs, and the results of the method in reference [20] are consistent with the results of the proposed method. Of course, there is still a certain gap between the segmentation results of the methods in reference [15, 20] and the proposed methods. There are still deficiencies in the processing of some details, which need to be improved.

It should be noted that in the image segmented by the proposed method, there are some scattered black areas, while the color images displayed by other methods have no black areas. This is because, in the annotation of the proposed method, the black areas represent the negligible parts, and the objects in this part are not added to the semantic segmentation task, which belongs to the object category other than the 19 categories in the dataset. In general, the results on the Cityscapes dataset show that the proposed method adaptively enhances the semantic information of low-level features and obtains better segmentation results through the combination of rough segmentation and fine segmentation.

**3.4. AUC Comparison of Several Detection Methods.** In order to compare the detection performance of the three detection methods on various scales numerically, the area under curve (AUC) enclosed by each curve and coordinate axis is calculated, and the calculation results are shown in Table 1.

As can be seen from Table 1, the detection performance of the three methods for medium-sized targets is better than that for small targets and large targets, which is due to the fact that the medium-sized targets in the Cityscapes dataset occupy most of the samples. In addition, the

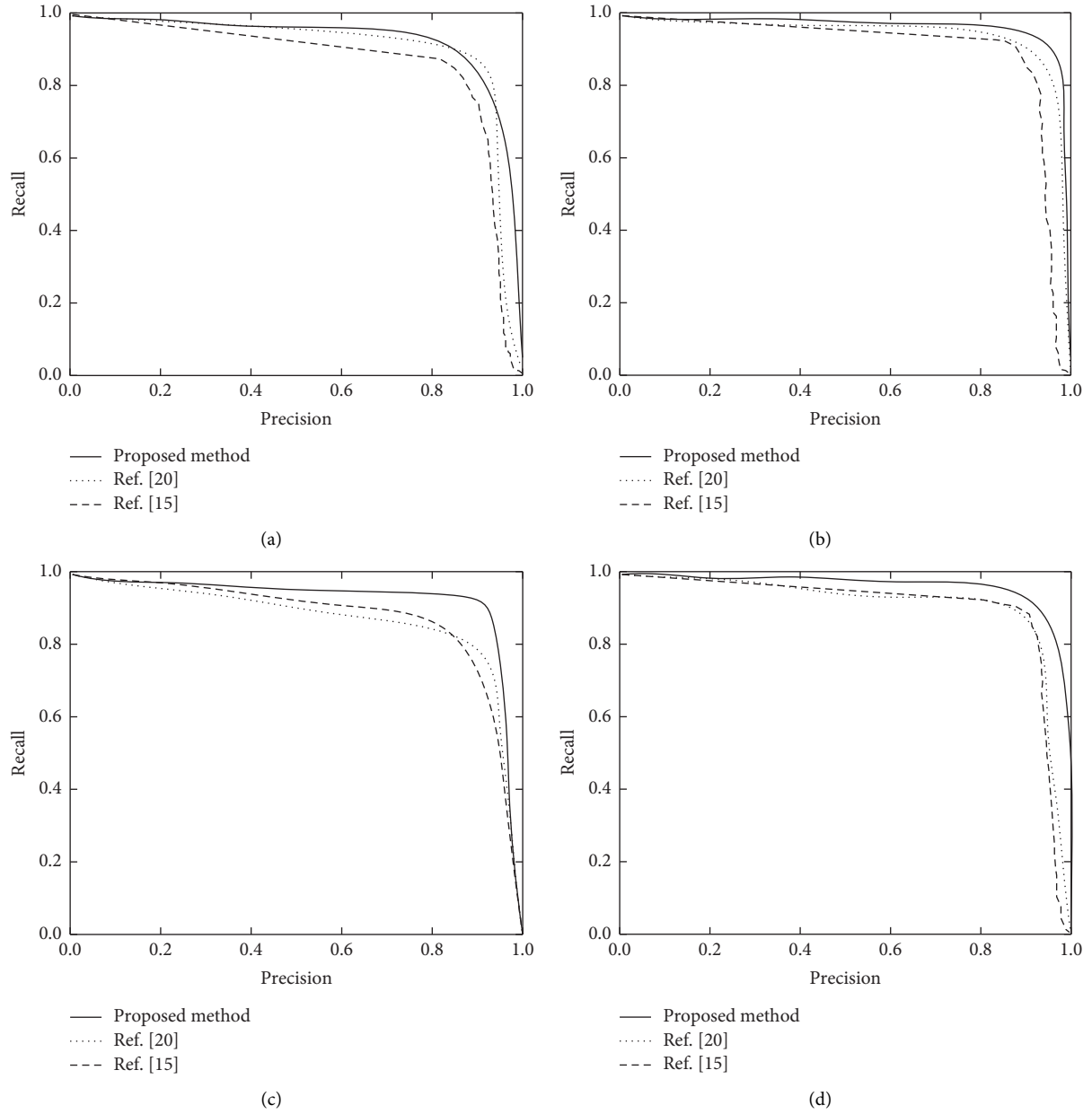


FIGURE 4: Comparison of P-R curves between the proposed method and other methods. (a) The target scale is [0, 64] pixels. (b) The target scale is [64, 192] pixels. (c) The target scale is [192, 500] pixels. (d) The target scale is [0, 500] pixels.

detection results of the proposed method on all scales are better than the other two detection methods, and the AUC of overall size target segmentation is 93.6%. Moreover, compared with reference [20], target detection results are improved by up to 5.4%, indicating that the combination of the fully convolution neural network and conditional

random field for rough and fine image segmentation can alleviate the multiscale problem in the target detection method. At the same time, compared with reference [15], the AUC of the proposed method in the segmentation of overall size targets is improved by 7.1%, and the segmentation effect is remarkable.



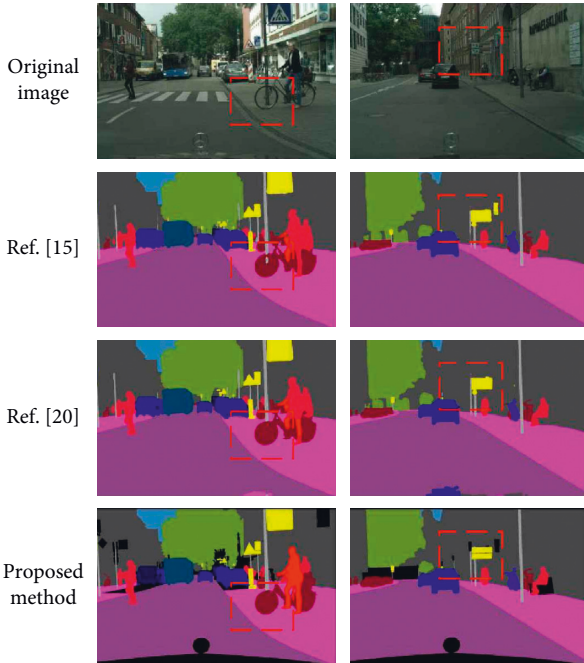


FIGURE 5: Comparison of visual segmentation results on Cityscapes dataset.

TABLE 1: AUC comparison of three methods.

Method	Ref. [15]	Ref. [20]	Proposed method
Small target	0.836	0.878	0.902
Medium target	0.871	0.903	0.957
Big target	0.908	0.896	0.949
Overall scale target	0.865	0.884	0.936

#### 4. Conclusion

At present, most of the existing image segmentation researches focus on integrating feature information of different depths to improve the performance of segmentation tasks. Some of them strengthen the features before feature fusion but focus on the overall enhancement of features and ignore the local differences of features. Therefore, an image semantic segmentation method based on the fully convolution neural network and conditional random field is proposed. Among them, the rough segmentation result of the image is obtained by DFN, which is taken as the first-order potential of the fully connected conditional random field. The spatial context information is introduced through the Gaussian second-order potential function to realize the fine segmentation of the image. The experimental results based on the Cityscapes dataset show that the proposed method can obtain multiscale features with the help of DFN, so its P-R curves are ideal under different pixels. Combined with the image fine segmentation results obtained by conditional random field, the AUC of the overall size target is 93.6%, and the segmentation results are more accurate.

The proposed method focuses on the improvement of accuracy without considering the real time. Although many

researchers have invested in the research of real-time image semantic segmentation, it still needs to be deeply explored from the practical application to ensure the reliability of recognition results and the security of the application. For example, in the automatic driving scene, not only the accuracy of the model and high recognition speed are required but also the application safety is still a major factor worthy of consideration.

#### Data Availability

The data used to support the findings of this study are included within the article.

#### Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

#### Acknowledgments

This work was supported by the Science and technology project of Henan Province (No. 202102210353).

#### References

- [1] W. Liu, "Real-time obstacle detection based on image semantic segmentation and fusion network," *Traitement du Signal*, vol. 38, no. 2, pp. 443–449, 2021.
- [2] C.-H. Xu, C. Shi, and Y.-N. Chen, "End-to-end dilated convolution network for document image semantic segmentation," *Journal of Central South University*, vol. 28, no. 6, pp. 1765–1774, 2021.
- [3] L. Mi and Z. Chen, "Corrigendum to "Superpixel-enhanced deep neural forest for remote sensing image semantic segmentation," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 168, no. 2, pp. 153–155, 2020.
- [4] L. Mi and Z. Chen, "Superpixel-enhanced deep neural forest for remote sensing image semantic segmentation," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 159, no. 9, pp. 140–152, 2020.
- [5] Y. Li, T. Shi, Y. Zhang, and W. Z. H. Chen, "Learning deep semantic segmentation network under multiple weakly-supervised constraints for cross-domain remote sensing image semantic segmentation," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 175, no. 6, pp. 20–33, 2021.
- [6] G. A. Qian and A. Qd, "Semantic image segmentation based on SegNetWithCRFs," *Procedia Computer Science*, vol. 187, no. 3, pp. 300–306, 2021.
- [7] S. Zhu, "Semantic segmentation of remote sensing image based on convolutional neural network," *Computer Science and Application*, vol. 11, no. 2, pp. 356–369, 2021.
- [8] Y. Zhu and M. Zhao, "Registration of laser point cloud and optical image in urban area based on semantic segmentation," *Acta Photonica Sinica*, vol. 50, no. 1, pp. 188–202, 2021.
- [9] S. Mamoon, M. A. Manzoor, F. E. Zhang, and Z. J.-f. Ali, "SPSSNet: a real-time network for image semantic segmentation," *Frontiers of Information Technology & Electronic Engineering*, vol. 21, no. 12, pp. 1770–1782, 2020.
- [10] M. Rezaei, H. Yang, and C. Meinel, "Recurrent generative adversarial network for learning imbalanced medical image semantic segmentation," *Multimedia Tools and Applications*, vol. 79, no. 21–22, pp. 15329–15348, 2020.

- [11] M. Z. Khan, M. K. Gajendran, Y. Lee, and M. A. Khan, "Deep neural architectures for medical image semantic segmentation: review," *IEEE Access*, vol. 9, no. 99, pp. 83002–83024, 2021.
- [12] A. Oluwasammi, M U. Aftab, Z. Qin, and S T Ngo, "Features to text: a comprehensive survey of deep learning on semantic segmentation and image captioning," *Complexity*, vol. 2021, Article ID 5538927, 19 pages, 2021.
- [13] B. Yan, X. Niu, B. Bare, and W. Tan, "Semantic segmentation guided pixel fusion for image retargeting," *IEEE Transactions on Multimedia*, vol. 22, no. 3, pp. 676–687, 2020.
- [14] T.-T. Nguyen-Quynh, S.-H. Kim, and N.-T. Do, "Image colorization using the global scene-context style and pixel-wise semantic segmentation," *IEEE Access*, vol. 8, no. 2, pp. 214098–214114, 2020.
- [15] F. Shen, Z. H. Wang, and Z. M. Lu, "Weakly supervised classification model for zero-shot semantic segmentation," *Electronics Letters*, vol. 56, no. 23, pp. 1247–1250, 2020.
- [16] X. Niu, B. Yan, W. Tan, and J. Wang, "Effective image restoration for semantic segmentation," *Neurocomputing*, vol. 374, no. 6, pp. 100–108, 2020.
- [17] K. Liu, Z. Ye, H. Guo, and D. L. F.-Y. Cao, "FISS gan: a generative adversarial network for foggy image semantic segmentation," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 8, pp. 1428–1439, 2021.
- [18] C. W. Weiwei, K. Wang, K. Wang, Z. Li, H. Li, and S. Liu, "Lane departure warning systems and lane line detection methods based on image processing and semantic segmentation: a review," *Journal of Traffic and Transportation Engineering*, vol. 42, no. 6, pp. 25–51, 2020.
- [19] J. Choi and B. Choi, "Highly contrast image correction for dim boundary separation of image semantic segmentation," *IEEE Access*, vol. 9, no. 6, pp. 64142–64152, 2021.
- [20] T. Yamane and P.-j. Chun, "Crack detection from a concrete surface image based on semantic segmentation using deep learning," *Journal of Advanced Concrete Technology*, vol. 18, no. 9, pp. 493–504, 2020.
- [21] J. Fan, Z. Zhang, T. Tan, and C. J. Song, "CIAN: cross-image affinity net for weakly supervised semantic segmentation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 7, pp. 10762–10769, 2020.
- [22] T. Wu, S. Tang, R. Zhang, and J. Y. Cao, "CGNet: a light-weight context guided network for semantic segmentation," *IEEE Transactions on Image Processing*, vol. 30, no. 2, pp. 1169–1179, 2021.
- [23] H. Ding, X. Jiang, B. Shuai, and A. Q. G. Liu, "Semantic segmentation with context encoding and multi-path decoding," *IEEE Transactions on Image Processing*, vol. 29, no. 3, pp. 3520–3533, 2020.
- [24] W. Shimoda and K. Yanai, "Weakly supervised semantic segmentation using distinct class specific saliency maps," *Computer Vision and Image Understanding*, vol. 191, no. 3, Article ID 102712, 2020.
- [25] S. Zhao, G. Hao, Y. Zhang, and S. Wang, "A real-time semantic segmentation method of sheep carcass images based on ICNet," *Journal of Robotics*, vol. 2021, no. 2, pp. 1–12, Article ID 8847984, 2021.
- [26] L. Zhang, X. Hu, Y. Zhou, and G. S. Zhou, "Memristive DeepLab: a hardware friendly deep CNN for semantic segmentation," *Neurocomputing*, vol. 451, no. 6, pp. 181–191, 2021.
- [27] I. Kotaridis and M. Lazaridou, "Remote sensing image segmentation advances: a meta-analysis," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 173, no. 3, pp. 309–322, 2021.

## Research Article

# WTD-PSD: Presentation of Novel Feature Extraction Method Based on Discrete Wavelet Transformation and Time-Dependent Power Spectrum Descriptors for Diagnosis of Alzheimer's Disease

Ali Taghvirashidizadeh <sup>1</sup>, Fatemeh Sharifi,<sup>2</sup> Seyed Amir Vahabi,<sup>3</sup> Aslan Hejazi,<sup>4</sup> Mehrnaz SaghbatTorbaty,<sup>5</sup> and Amin Salih Mohammed<sup>6,7</sup>

<sup>1</sup>Islamic Azad University, Central Tehran Branch (IAUCTB), Department of Electrical and Electronics Engineering, Tehran, Iran

<sup>2</sup>Department of Electrical Engineering, University of Applied Science and Technology, Bushehr, Iran

<sup>3</sup>Department of Computer Engineering, Deylaman Institute of Higher Education, Lahijan, Iran

<sup>4</sup>Department of Electrical Engineering, Islamic Azad University, Science and Research Branch, Tehran, Iran

<sup>5</sup>Department of Biomedical Engineering, Faculty of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran

<sup>6</sup>Department of Computer Engineering, College of Engineering and Computer Science, Lebanese French University, Erbil, Kurdistan Region, Iraq

<sup>7</sup>Department of Software and Informatics Engineering, Salahaddin University, Erbil, Kurdistan Region, Iraq

Correspondence should be addressed to Ali Taghvirashidizadeh; [ali.taghavi.eng@iauctb.ac.ir](mailto:ali.taghavi.eng@iauctb.ac.ir)

Received 4 December 2021; Revised 2 February 2022; Accepted 25 February 2022; Published 11 May 2022

Academic Editor: Mohammed A. A. Al qaness

Copyright © 2022 Ali Taghvirashidizadeh et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Alzheimer's disease (AD) is a type of dementia that affects the elderly population. A machine learning (ML) system has been trained to recognize particular patterns to diagnose AD using an algorithm in an ML system. As a result, developing a feature extraction approach is critical for reducing calculation time. The input image in this article is a Two-Dimensional Discrete Wavelet (2D-DWT). The Time-Dependent Power Spectrum Descriptors (TD-PSD) model is used to represent the subbanded wavelet coefficients. The principal property vector is made up of the characteristics of the TD-PSD model. Based on classification algorithms, the collected characteristics are applied independently to present AD classifications. The categorization is used to determine the kind of tumor. The TD-PSD method was used to extract wavelet subbands features from three sets of test samples: moderate cognitive impairment (MCI), AD, and healthy controls (HC). The outcomes of three modes of classic classification methods, including KNN, SVM, Decision Tree, and LDA approaches, are documented, as well as the final feature employed in each. Finally, we show the CNN architecture for AD patient classification. Output assessment is used to show the results. Other techniques are outperformed by the given CNN and DT.

## 1. Introduction

The brain is the body's most important organ. The disorders that affect the brain are extremely important to manage since, in most situations, once alterations occur, they are irreversible in extreme circumstances. Dementia is defined as the loss of cognitive and functional thinking abilities. The most prevalent cause of dementia is AD. The AD strikes people in their mid-60s. Alzheimer's disease affects more

than 5.5 million individuals worldwide [1]. Memory loss, language problems, and behavioral changes are all indications of AD. Symptoms of the nonmemory part include trouble locating words, eye problems, decreased cognition, and poor judgment. Brain imaging, cerebrospinal fluid, and blood are the biological signs. Normal age-related decrease in cognitive function, which is more gradual and associated with less impairment, should be distinguished from AD. The disease frequently begins with little symptoms and

progresses to serious brain damage. Dementia affects people differently; therefore their abilities deteriorate at varying rates. Early and reliable identification of AD is advantageous to disease management. Neuroimaging techniques like magnetic resonance imaging (MRI) and computed tomography (CT), as well as single-photon emission computed tomography (SPECT) and positron emission tomography (PET), can be utilized to rule out other forms of dementia or subtypes. It has the potential to forecast the progression of prodromal into AD. Neurologists can use medical image processing and machine learning methods to see if a person is developing AD. Image segmentation and classification are critical tasks in MRI data analysis for detecting AD [2]. Structural MRI (SMRI) provides visual information regarding the atrophic areas of the brain caused by the tissue level abnormalities that underpin AD/MCI. PET measures cerebral glucose metabolism, which is a reflection of functional brain activity [3]. The quantity of amyloid beta-protein and amyloid tau tangles accumulated in the cerebrospinal fluid (CSF) is an early predictor of AD. SMRI has already been shown to be sensitive to presymptomatic illness and might be used as a disease biomarker [4]. MRI appears to be the most sensitive imaging examination of the brain in everyday clinical practice. It provides information on gray matter, white matter, and CSF morphology. Structural MRI can record atrophic brain areas non-invasively, allowing us to see anatomical alterations in the brain. As a result, they have been recognized as a possible indication of illness development, and ML approaches for disease detection are being researched extensively [5].

The MRI scan can be utilized in image processing to evaluate the likelihood of early detection of AD. Intensity adjustment,  $K$ -means clustering, and the region growing method are image processing techniques used in MRI to extract white and gray matter [6]. The same approach may be used to compute brain volume. Because the raw MRI brain image is too large to be utilized for classification, the MR images must be preprocessed before feature extraction and classification can be performed for illness diagnosis. Through the warping of labeled atlas, one of the most generally used approaches is to divide the image into numerous anatomical areas, that is, regions of interest (ROIs), and the regional measurements, such as volumes, are calculated as the features for AD classification [7]. To identify the most discriminative features from ROIs for multi-modality classification of AD/MCI, a discriminative multitask algorithm was presented. In ML, each data item should be characterized as a feature vector.

There are numerous research advocated extracting various characteristics from MRI scans and then classifying the resulting vectors. The quality of the produced feature vectors is, nevertheless, reliant on image preprocessing due to registration errors and noise. As a result, domain knowledge is required to extract discriminative features. CNN's layered design has a big influence on its performance. Greater classification accuracy is anticipated to arise from a layer structure that is better suited for MRI images. The input images in this article are Two-Dimensional Discrete Wavelets (2D-DWT). The Time-Dependent Power

Spectrum Descriptors (TD-PSD) model is used to represent the subbanded wavelet coefficients. The primary property vector is made up of the characteristics of the TD-PSD model. Based on classification algorithms, the collected characteristics are applied independently to present AD classifications. The classification is used to determine the kind of tumor. For feature extraction of wavelet subbands from three sets of mild cognitive impairment (MCI), AD, and HC test data, we employed the TD-PSD technique.

## 2. Literature Review

For diagnosing AD, feature vectors from MRI images must be extracted. Several feature extraction techniques have been proposed in the recent decade since the outcome of ML is determined by the extracted feature vectors. Employing many specified templates, Liu et al. [8] retrieved multiview feature representations for subjects and divided subjects within a particular class into distinct subclasses in each view space. Support vector machine-based (SVM) ensemble learning was used. Suk et al. proposed a multitask and multikernel SVM learning approach for a stacked autoencoder with a deep-learning-based feature representation [9]. Due to registration mistakes and noise, the quality of the recovered features is dependent on image preprocessing. As a result, domain knowledge is required while extracting discriminative features. It takes a long time and a lot of effort to acquire hand-crafted features. More crucially, hand-crafted features seldom generalize well. As a consequence, this study recommends employing deep learning to extract data characteristics. Sadeghipour and Sahragard [10] developed a novel approach for facial identification that is based on an enhanced SIFT algorithm. Acharya et al. [11] created an ML system that can detect AD symptoms in a brain scan. For classification, the system combined MRI with a variety of feature extraction techniques. The T2 imaging sequence was used to get the images. Filtering, feature extraction, Student's  $t$ -test-based feature selection, and  $k$ -Nearest Neighbor- (KNN-) based classification were among the quantitative approaches used in the paradigm. The findings revealed that when compared to other approaches, the Shearlet Transform (ST) feature extraction methodology provides better outcomes for Alzheimer's diagnosis. With the ST+KNN approach, the suggested tool achieved 94.54 percent accuracy, 88.33 percent precision, 96.30 percent sensitivity, and 93.64 percent specificity. According to Sadeghipour et al. [12], combining fireflies with intelligent systems would lead to breast cancer detection. The results show that by comparing the performance of the suggested system to other methods, it is evident that it is superior in both performance and accuracy. Sadeghipour and Moradisabzevar [13] investigated the development of intelligent toy cars as a method of screening children with autism. The results show that the screening of autistic children was 100 percent accurate. The study by Zhou et al. [14] investigated probabilistic inflection points for the decomposition of LiDAR hidden echo signals. Yan et al. [15] examined the structure and in vitro test results of waxy and regular maize starches after thermal processing using plasma-activated water. Eslami and Yun [16] have developed a novel approach called A + MCNN

and have compared it to four commonly used deep classifiers in the transportation domain as well as the standard CNN classifier. Sadeghipour and Hatam [17] developed the XCSLA System to help in the diagnosis of diabetes. Hassantabar et al. [18] implemented three deep learning-based methods using X-ray images of the lungs to detect and diagnose COVID-19 patients. According to Sadeghipour and Hatam [19], the Expert Clinical System for Diagnosing Obstructive Sleep Apnea with Help from the XCSR Classifier helps diagnose this sleep disorder. Abadi et al. [20] have proposed a hybrid swarm algorithm and genetic algorithm (HSSAGA) model for solving nurses' scheduling and designation issues. In comparison with state-of-the-art approaches, this algorithm outperforms the suggested test function algorithm. Odusami et al. [21] suggested a deep-learning-based technique for predicting MCI, early MCI (EMCI), late MCI (LMCI), and AD. On the EMCI versus AD, LMCI versus AD, and MCI versus EMCI classification scenarios, the fine-tuned ResNet18 network achieved classification accuracy of 99.99%, 99.95%, and 99.95%, respectively, on the fine-tuned ResNet18 network. In terms of accuracy, sensitivity, and specificity, the suggested model outperformed other well-established models in the literature. Sharifi et al. [22] described a technique for diagnosing weary and exhausted feet using digital footprint photos. The current CNN technique outperforms existing methods and may be employed in the development of future fatigue detection systems. Furthermore, a conclusion neural network can be applied to the detection of tumors [23], the scheduling problems for health care systems [24], and the optimization of users based on a clustering method [25]. A new approach to penetration testing based on extended classifier networks has been proposed by Yazdani et al. [26]. A model of an application created for mobile Android systems was provided by Lauraitis et al. [27], which may be used to examine central nervous system movement problems occurring in individuals suffering from Huntington's, Alzheimer's, or Parkinson's illnesses. Specifically, the model detects tremors as well as cognitive deficits through the use of touch and visual stimulation modalities, among other things. According to the findings, the adoption of intelligent applications that may assist in the evaluation of neurodegenerative illnesses is a significant advancement in medical diagnostics and should be encouraged. According to Sadeghipour et al. [28], the xcsla system can be used to develop an intelligent diabetes diagnosis system. According to the results of the program implementation document (pid) on databases, the proposed technique can detect diabetes more accurately than the conventional xcs system, the Elman neural network, svm clustering, knn, c4.5, and ad tree. Farahanipad et al. [29] developed a pipeline for the identification of hand 2D keypoints using unpaired image-to-image translation. In Shi et al.'s [30] study, they investigated the effect of ultrasonic intensity on the structure and characteristics of sago starch complexes and their implications for the quality of Chinese steamed bread. Sadeghipour et al. [31] developed a new expert clinical method for the diagnosis of obstructive sleep apnea using the XCSR classifier. Rezaei et al. [32] used depth images to automate mild segmentation of hand parts. According to the results, a model without segmentation-based labels may

achieve a mIoU of 42%. Quantitative and qualitative findings support our method's efficiency.

Yue et al. [33] use automated anatomical labeling (AAL) template to divide the brain into 90 regions of interest (ROIs). They choose the informative voxels in each ROI with a baseline of their values and arrange them into a vector to divide the uninformative data. The first stage characteristics were then chosen based on the voxel correlation between distinct groups. The fetched voxels were then put into a convolutional neural network (CNN) to understand the profoundly hidden properties of each subject's brain features maps. The testing findings showed that the suggested technique was reliable and had a promising performance when compared to other methods in the literature.

For increasing classification accuracy and identifying high-order features that potentially provide pathological information, Li et al. [44] used a novel feature extraction approach known as radiomics. As a consequence, they defined ROIs as brain regions mostly dispersed in the temporal, occipital, and frontal areas. A total of 168 radiomic characteristics of Alzheimer's disease were found to be stable ( $\alpha > 0.8$ ). The maximum accuracies for categorizing AD versus HC, MCI versus HCs, and AD versus MCI were 91.5 percent, 83.1 percent, and 85.9 percent, respectively, in the classification trial. Silva et al. [46] suggested a model for diagnosing AD based on deep feature extraction for MRI classification. The goal of this model was to distinguish between AD and HC. For extracting the best characteristics of the selected region, the CNN architecture was also developed in three convolutional layers. The model's effectiveness and reliability for the diagnosis of AD were shown by a comparison study with previous studies in the literature. Table 1 lists several more techniques.

### 3. Methods and Materials

*3.1. Discrete Wavelet Transform (DWT).* If  $f(x) \in L^2(R)$  is a wavelet expansion function that is connected to wavelet  $\psi(x)$  and scaling  $\varphi(x)$ , we get [47]

$$f(x) = \sum_k c_{j_0}(k) \varphi_{j_0,k}(x) + \sum_{j=j_0}^{\infty} \sum_k d_j(k) \psi_{j,k}(x). \quad (1)$$

$c_{j_0}(k)$ 's are scaling coefficients, and  $j_0$  is a starting counter. The  $d_j(k)$  coefficients are wavelet coefficients (see Figure 1). The following are the expansion coefficients:

$$c_{j_0}(k) = \langle f(x)t, n\tilde{\varphi}_{j_0,k}q(x) \rangle = \int f(x)\tilde{\varphi}_{j_0,k}(x)dx, \quad (2)$$

$$d_j(k) = \langle f(x)t, n\tilde{\psi}_{j,k}q(x) \rangle = \int f(x)\tilde{\psi}_{j,k}(x)dx, \quad (3)$$

$$\begin{aligned} \tilde{\varphi}_{j,k}(x) &= 2^{j/2}\tilde{\varphi}(2^j n - k), \\ \tilde{\psi}_{j,k}(x) &= 2^{j/2}\tilde{\psi}(2^j n - k). \end{aligned} \quad (4)$$

It is also known as the discrete wavelet transform of  $f(x)$  if the expansion function is a series of crisp numbers.

TABLE 1: Summary of method for diagnosis of Alzheimer using computer-aided approach.

Author	Year	Data	Classes	Feature extraction	Classifier
Yang et al. [34]	2021	Magnetoencephalography	AD, MCI, HC	Space–frequency–time domain feature extraction	3-NN and QBNC
Hedayati et al. [35]	2021	3D-MRI	AD, MCI, HC	Ensemble of pre-trained auto encoder	CNN
Biagetti et al. [36]	2021	EEG signal	AD, HC	Robust-PCA	KNN, DT, SVM, NB
Chen and Xia [37]	2021	sMRI	AD, MCI, HC	Deep feature extraction	CNN
Ahmadi et al. [38]	2021	MRI	Low, mild, moderate, severe stage	Brain tumor diagnosis	Fuzzy logic, CNN
Amini et al. [39]	2021	fMRI	Low, mild, moderate, severe stage	Robust multitask feature extraction method	KNN, SVM, DT, LDA, CNN
Janghel and Rathore [40]	2020	fMRI, PET	AD, HC	Image map	SVM, DT, LDA, CNN
Ahmadi et al. [41]		MRI	Low, mild, moderate, severe stage	Tumor area segmentation	QAIS-DSNN
Parmar et al. [42]	2020	fMRI	AD, EMCI, LMCI, HC	Spatiotemporal feature extraction	3D-CNN
Ahmadi et al. [43]	2021	MRI	Low, mild, moderate, severe stage	Brain tumor diagnosis	CNN
Li et al. [44]	2019	18F-FDG PET imaging	AD, MCI, HC	High-order radiomic features extraction	SVM
Yue et al. [33]	2019	MRI	AD, MCI, HC	Voxel-based hierarchical method	CNN
Acharya et al. [11]	2019	MRI	AD, HC	Shearlet transform, curvelet, contourlet, complex wavelet, discrete wavelet, empirical wavelet, dual tree complex wavelet	KNN
Fiscon et al. [45]	2018	EEG signal	AD, MCI, HC	Fast Fourier transform, discrete wavelet transform	DT

The expansion series is represented by equations (2) and (3) (DWT pair) [47, 48]:

$$\begin{aligned}
W_\varphi(j_0, k) &= \frac{1}{\sqrt{M}} \sum_{x=0}^{M-1} f(x) \bar{\varphi}_{j_0, k}(x), \\
W_\psi(j, k) &= \frac{1}{\sqrt{M}} \sum_{x=0}^{M-1} f(x) \bar{\psi}_{j, k}(x) \quad j \geq j_0, \\
f(x) &= \frac{1}{\sqrt{M}} \sum_k W_\varphi(j_0, k) \varphi_{j_0, k}(x) + \frac{1}{\sqrt{M}} \sum_{j=j_0}^{\infty} \sum_k W_\psi(j, k) \psi_{j, k}(x),
\end{aligned} \tag{5}$$

where  $f(x)$ ,  $\varphi_{j_0, k}(x)$ , and  $\psi_{j, k}(x)$  are discrete variables,  $x=0, 1, \dots, M-1$ ,  $j=0, 1, \dots, J-1$ ,  $k=0, 1, 2, \dots, M-1$  functions, where  $M$  is the number of samples to be converted, and  $J$  is the number of transformation levels; it equals  $2^J$ . To construct a 1D scaling function  $\phi$  and associated wavelet  $\psi$  [39], 2D,  $\varphi(x, y)$ , and 3D,  $\psi^H(x, y)$ ,  $\psi^V(x, y)$ , and  $\psi^D(x, y)$ , are usually necessary.

$$\begin{aligned}
\varphi(x, y) &= \varphi(x)\varphi(y), \\
\psi^H(x, y) &= \psi(x)\varphi(y), \\
\psi^V(x, y) &= \varphi(y)\psi(x), \\
\psi^D(x, y) &= \psi(x)\psi(y).
\end{aligned} \tag{6}$$

A two-level wavelet transformation creates four subbands, as seen in Figure 1. In this diagram  $2\downarrow$ ,  $\psi^H$ ,  $\psi^V$ , and  $\psi^D$  indicated deviations along horizontal, vertical, and

diagonal boundaries, respectively. Digital filtration and downsamplers can be used to perform 2D-DWT. The additional subbands are produced using discrete 2D scaling functions and 1D-FWT on  $f(x, y)$  [49].

**3.2. Feature Extraction.** The discrete Fourier transform (DFT) is supposed to explain the signal trace as a function of frequency  $X[k]$  as a product of the sampled representation of the signal as  $x[j]$  with  $j=1, 2, \dots, N$ , length  $N$ , and sampling frequency  $f_s$  Hz. If we remember Parseval's theorem, the sum of the square of the function equals the whole square of its transformation. We begin the feature extraction procedure.

$$\sum_{j=0}^{N-1} |x[j]|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X[k]X^*[k]| = \sum_{k=0}^{N-1} P[k], \tag{7}$$

$P[k]$  is the phase-excluded power spectrum, according to the preceding equation. This implies that multiplying  $X[k]$  by the  $X^*[k]$  conjugate divided by  $N$  yields the frequency index.

$P[k]$  is the phase-excluded power spectrograph; that is,  $X[k]$  has its conjugate  $X^*[k]$ , which is separated by  $N$ , which is compounded by  $k$ , and frequency index. The Fourier transform's whole notion of frequency is usually thought to be symmetrical with respect to zero frequency. It has similar sections that cover both positive and negative

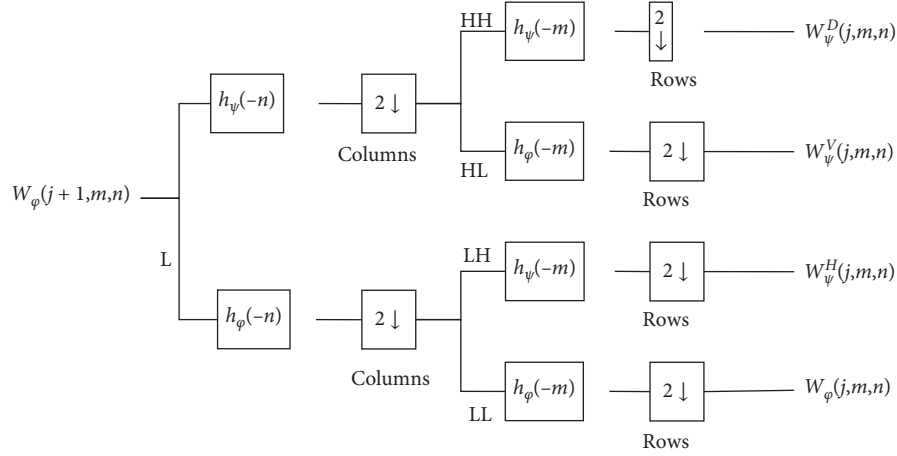


FIGURE 1: The two-dimensional DWT diagram.

frequencies. This symmetry does not apply to all frequencies in the spectrum, including positive and negative ones. Because we do not have comprehensive accessibility to the time domain, we cannot employ spectral power from there. By the statistical approach of the frequency distribution model, all irregular moments are also zero, corresponding to the concept of a one-minute  $m$  of the power spectral density of order  $n$   $P[k]$ .

$$m_n = \sum_{k=0}^{N-1} k^n P[k]. \quad (8)$$

The Parseval theorem might indeed be used when  $n = 0$  is used. For nonzero values of  $n$ , the Fourier transform time-differentiation feature could be applied. The  $n^{\text{th}}$  means multiplying the  $k$  by the spectrum to the  $n^{\text{th}}$  power, according to this feature. The derivative of a time-domain function is alluded to as  $\Delta^n$  for distinct time signals.

$$F[\Delta^n x[j]] = k^n X[k]. \quad (9)$$

*Root Squared Zero-Order Moment ( $\overline{m_0}$ ).* This is a function that displays the frequency domain's total power and looks like this

$$\overline{m_0} = \sqrt{\sum_{j=0}^{N-1} x[j]^2}. \quad (10)$$

All channels could standardize their related zero-order moments by splitting them into zero-order moments.

*Root Squared Second- and Fourth-Order Moments.* The second time is utilized as power, but it is subsequently shifted to  $k^2 P[k]$ , which refers to the frequency function:

$$\overline{m_2} = \sqrt{\sum_{k=0}^{N-1} k^2 P[k]} = \sqrt{\frac{1}{N} \sum_{k=0}^{N-1} (kX[k])^2} = \sqrt{\sum_{j=0}^{N-1} (\Delta x[j])^2}. \quad (11)$$

The moment is obtained by repeating this approach:

$$\overline{m_4} = \sqrt{\sum_{k=0}^{N-1} k^4 P[k]} = \sqrt{\sum_{k=0}^{N-1} (\Delta^2 x[j])^2}. \quad (12)$$

The overall energy of the signal is reduced when the second and fourth signals are taken into account. For decreasing the noise impact on all moment-based features, to normalize the domains of  $m_0$ ,  $m_2$ , and  $m_4$ , we perform the following power transformation:

$$m_0 = \frac{\overline{m_0}^\lambda}{\lambda}, m_2 = \frac{\overline{m_2}^\lambda}{\lambda}, m_4 = \frac{\overline{m_4}^\lambda}{\lambda}. \quad (13)$$

The experimental value of  $\lambda$  is set to 0. As a result of these settings, the first three features extracted are as follows:

$$\begin{aligned} f_1 &= \log(m_0), \\ f_2 &= \log(m_0 - m_2), \\ f_3 &= \log(m_0 - m_4). \end{aligned} \quad (14)$$

*Sparseness.* This feature calculates the quantity of vector energy in a small number of additional components. It is then followed by

$$f_4 = \log\left(\frac{m_0}{\sqrt{m_0 - m_2} \sqrt{m_0 - m_4}}\right). \quad (15)$$

A feature shows a vector with all elements equivalent to a zero-sparseness index, i.e.,  $m_2$  and  $m_4 = 0$ , due to differentiation and  $\log(m_0/m_0) = 0$ . All other sparseness levels, on the other hand, should have a value greater than zero.

*Irregularity Factor (IF).* The ratio of peak numbers divided by zero-crossings up is expressed by this metric. A random signal's number of upward zero-crossings (ZC) and number of peaks (NP) can only be characterized in terms of spectral instances. The following is how the appropriate feature should be written:

$$f_5 = \frac{ZC}{NP} = \frac{\sqrt{m_2/m_0}}{\sqrt{m_4/m_2}} = \sqrt{\frac{m_2^2}{m_0 m_4}} = \frac{m_2}{\sqrt{m_0 m_4}}. \quad (16)$$

*Covariance (COV).* Our COV function is described as the standard deviation on arithmetic averages divided by the standard deviation on arithmetic averages:

$$f_6 = \log\left(\frac{\sqrt{\sum_{j=0}^{N-1} (x - \bar{x})^2/n}}{\bar{x}}\right). \quad (17)$$

*Teager energy operator (TEO).* It mainly depicts the signal amplitude and instantaneous fluctuations, which are particularly sensitive to even little variations. TEO has been proposed as a method for modeling nonlinear speech signals. It was later widely employed in the audio signal processing industry. It is made up of the following parts:

$$f_7 = \log(\Psi(x[j])) = \log\left(\sum_{j=0}^{N-1} x^2[j] - x[j-1]x[j+1]\right). \quad (18)$$

**3.3. Proposed Feature Extraction Methods.** The goal of this research is to apply machine learning algorithms to identify Alzheimer's disease. Figure 2 show the block diagram of the proposed method. To begin, we employed a two-stage 2D-DWT to break down input images into wavelet subbands. The obtained wavelet coefficients are utilized to derive classification features. The TD-PSD model is then used to extract features, with the first step using HH1, HL1, LH1, and the second stage using LL2, HH2, HL2, and LH2. The PCA approach is employed to diminish the number of features, and then AD is categorized using multiple machine learning algorithms using the retrieved feature. The following is the pseudocode for the provided method (Algorithm 1).

## 4. Results

**4.1. Data Collection.** In AD, structural MR imaging results demonstrated microscopic neurodegeneration and are a measure of brain atrophy (loss of synapses, dendritic processes, and neurons). In volumetric or voxel-based assessments of brain atrophy, the degree of atrophy and the extent of cognitive impairment are closely associated. There is a relationship between cognitive decline and brain atrophy. Atrophy does not appear to be exclusive to AD on MR images. The degree of hippocampal atrophy, on the other hand, is highly correlated with autopsy Braak staging [50]. Braak staging of neurofibrillary tangles in ante-mortem MR imaging and postmortem AD staging match to the topographic distribution of atrophy on MR images (medial, basal, and lateral temporal lobes, as well as the medial parietal cortex) [51]. The data collection includes atrophy and clinical stages of AD. There is negligible atrophy in the cognitively normal control individual, while there is significant atrophy in the AD patient. The MCI

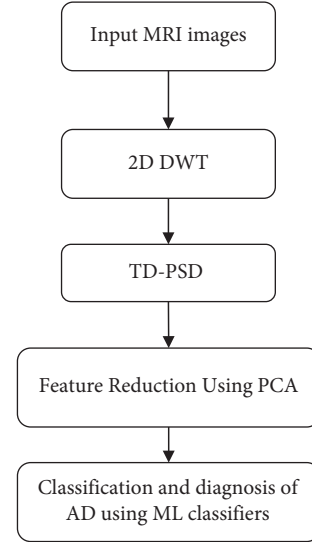


FIGURE 2: The block diagram of the proposed method.

individual, on the other hand, has an intermediate amount of atrophy. On Kaggle [52], the dataset is accessible online. The MRI images are  $256 \times 256$  PNG grayscale images that have been utilized to analyze and evaluate AD in three classes: AD, MCI, and an HC group.

**4.2. Feature Extraction and Reduction.** In this section, the process of feature extraction is described. Based on the conceptual diagram of Figure 2 and pseudocode, the first step in the presented method is wavelet decomposition. The results of decomposition are presented in Figure 3. Regarding Figure 3, a two-level decomposition is done for each input image. From the first step, three subbands of low-high (LH1), high-low (HL1), high-high (HH1), and from the second step low-low (LL2), LH2, HL2, and HH2 are used for feature extraction.

In the next step, each subband matrix is reshaped to a vector, and all the zeros are removed from the vectors. The final vectors are our pseudotime series that are used for feature extraction. The properties of the seven subbands are presented in Figure 4. Based on the amplitude and frequency of subbands, the LL2 subbands include the maximum number of points and properties of input images. However, all subbands are consequential in this diagnosis.

Based on the feature extraction results, each image has 49 features (7 subbands with 7 TD-PSD features). Moreover, principal component analysis is employed to reduce the features. Based on Figure 5, the first seven features include almost 100% effect of all features. Consequently, the number of features is reduced to 7 based on the screen plot in Figure 5(a). Moreover, the cumulative value of the eigenvalue is presented in Figure 5(b).

**4.3. Results of Classification.** In this section, the classification is done using different machine learning methods. The input layer of the classification methods is seven reduced features of the images, and the output layer is the three-class label of



**Input:**  $I_{m \times m} = \{m \times m\} \in R^2$   
**Output:** Labels (AD, MCI, HC)  
**Wavelet Decomposition:**  
 $[LL\ 1_{m/2 \times m/2}, LH\ 1_{m/2 \times v}, HL\ 1_{m/2 \times m/2}, HH\ 1_{m/2 \times m/2}] = 2\ D\_DWT(I_{m \times m})$   
 $[LL\ 2_{m/4 \times m/4}, LH\ 2_{m/4 \times m/4}, HL\ 2_{m/4 \times m/4}, HH\ 2_{m/4 \times m/4}] = 2D_{DWT}(LL\ 1_{m/4 \times m/4})$   
**Reshape sub-bands and remove zeros:**  
 $LH\ 1_v = \text{Reshape}(LH\ 1, [m/2 \times m/2, 1]) \& \text{Remove}(\text{zeros})$   
 $HL\ 1_v = \text{Reshape}(HL\ 1, [m/2 \times m/2, 1]) \& \text{Remove}(\text{zeros})$   
 $HH\ 1_v = \text{Reshape}(HH\ 1, [m/2 \times m/2, 1]) \& \text{Remove}(\text{zeros})$   
 $LL\ 2_v = \text{Reshape}(LL_2, [m/4 \times m/4, 1]) \& \text{Remove}(\text{zeros})$   
 $LH\ 2_v = \text{Reshape}(LH\ 2, [m/4 \times m/4, 1]) \& \text{Remove}(\text{zeros})$   
 $HL\ 2_v = \text{Reshape}(HL\ 2, [m/4 \times m/4, 1]) \& \text{Remove}(\text{zeros})$   
 $HH\ 2_v = \text{Reshape}(HH\ 2, [m/4 \times m/4, 1]) \& \text{Remove}(\text{zeros})$   
**Feature Extraction:**  
 $(f_1^{(1)}, f_2^{(1)}, f_3^{(1)}, f_4^{(1)}, f_5^{(1)}, f_6^{(1)}, f_7^{(1)}) = \text{TD\_PSD}(LH\ 1_v)$   
 $(f_1^{(2)}, f_2^{(2)}, f_3^{(2)}, f_4^{(2)}, f_5^{(2)}, f_6^{(2)}, f_7^{(2)}) = \text{TD\_PSD}(HL\ 1_v)$   
 $(f_1^{(3)}, f_2^{(3)}, f_3^{(3)}, f_4^{(3)}, f_5^{(3)}, f_6^{(3)}, f_7^{(3)}) = \text{TD\_PSD}(HH\ 1_v)$   
 $(f_1^{(4)}, f_2^{(4)}, f_3^{(4)}, f_4^{(4)}, f_5^{(4)}, f_6^{(4)}, f_7^{(4)}) = \text{TD\_PSD}(LL\ 2_v)$   
 $(f_1^{(5)}, f_2^{(5)}, f_3^{(5)}, f_4^{(5)}, f_5^{(5)}, f_6^{(5)}, f_7^{(5)}) = \text{TD\_PSD}(LH\ 2_v)$   
 $(f_1^{(6)}, f_2^{(6)}, f_3^{(6)}, f_4^{(6)}, f_5^{(6)}, f_6^{(6)}, f_7^{(6)}) = \text{TD\_PSD}(HL\ 2_v)$   
 $(f_1^{(7)}, f_2^{(7)}, f_3^{(7)}, f_4^{(7)}, f_5^{(7)}, f_6^{(7)}, f_7^{(7)}) = \text{TD\_PSD}(HH\ 2_v)$   
 $F_{49 \times 1} = f_j^{(i)}, \quad i = 1, 2, \dots, 7; \quad j = 1, 2, \dots, 7$   
**Feature reduction:**  
 $F' = \text{PCA}(F)$   
**Classification:**  
 Train (KNN ( $F'$ ), Labels)  
 Train (SVM ( $F'$ ), Labels)  
 Train (LDA ( $F'$ ), Labels)  
 Train (DT ( $F'$ ), Labels)  
 Train (CNN ( $F'$ ), Labels)  
**Performance Analysis:**  
 Plot (Confusion Matrix);  
 Plot (Performance Plot)  
 Plot (ROC)

ALGORITHM 1: The pseudocode for the proposed method.

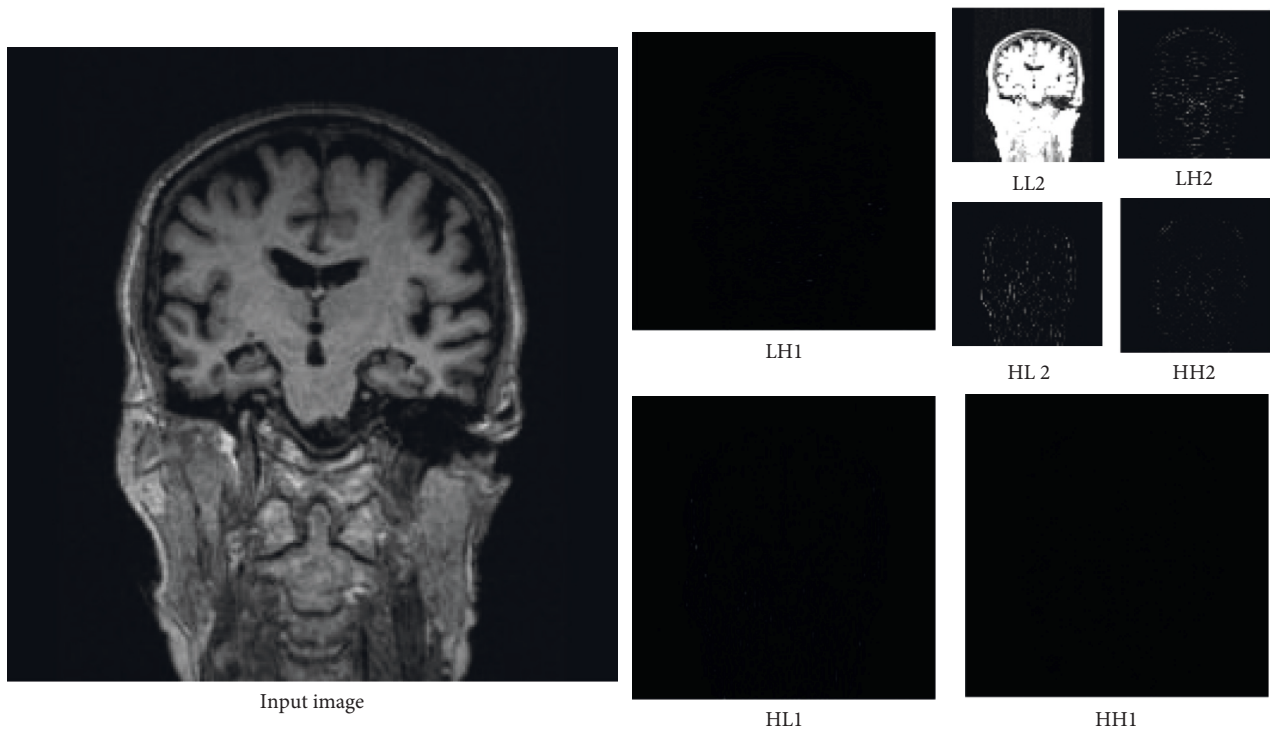


FIGURE 3: The subbands of the discrete wavelet transformation for an input image. HH: high pass-high pass subband, HL: high pass-low pass subband, LH: low pass-high pass, LL: low pass-low pass, 1: first level, and 2: second level.

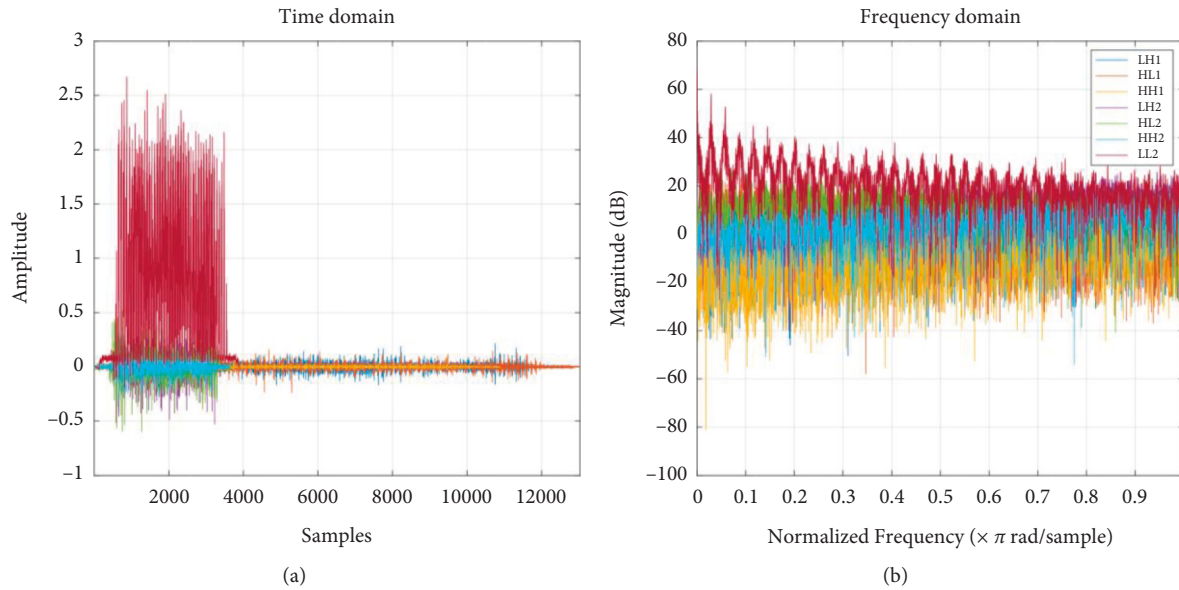


FIGURE 4: The frequency (a) and the amplitude (b) of the subbands for DWT for and input image.

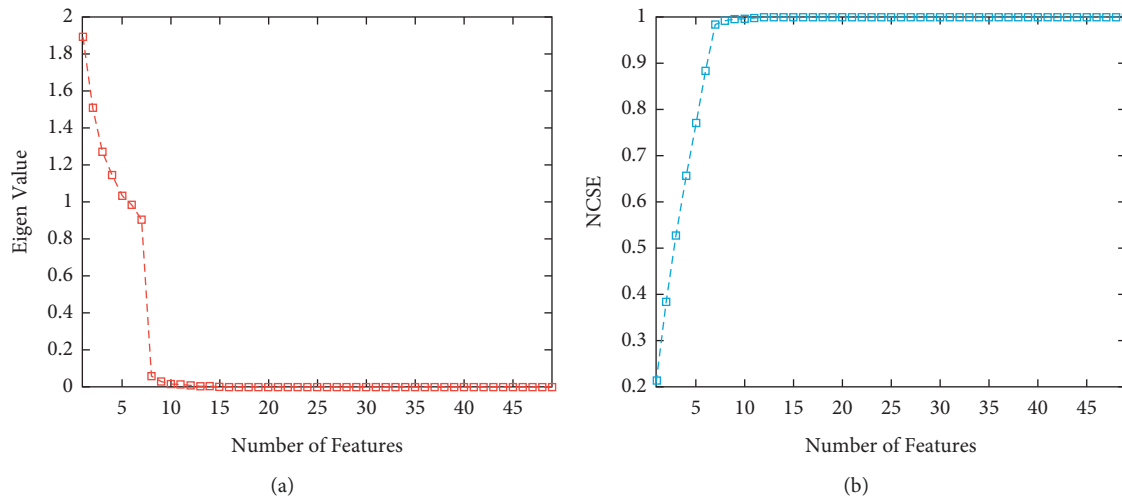


FIGURE 5: Results of feature reduction based on PCA method. (a) Eigen value of the features. (b) Normalized cumulative summation of Eigen values.

AD, MCI, and HC. Total 600 MRI images are employed for the classification of AD. The confusion matrixes of the presented methods are illustrated in Figure 6. The blue balls show the true values, and the red balls are the false value of the classification. Moreover, labels 1, 2, and 3 display the HC, AD, and MCI, respectively. Regarding the results of the KNN method, from 200 input HC, AD, and MCI images 193, 141, and 109 are diagnosed correctly. Based on the results, the sensitivity of the KNN for diagnosing Alzheimer's disease for HC is acceptable. Depending on the results, the SVM and LDA approaches reached the weak result for the diagnosis of AD. However, the results of DT show that the sensitivity of the method is 94%, 91.5%, and 97.5%, respectively. It means that the WTD-PSD is compatible with the DT approach for this problem. In other

words, 188, 183, and 195 MRI images from HC, AD, and MCI are detected, respectively. Moreover, the precision of the method is 91.70%, 95.30%, and 96.10% for HC, AD, and MCI, accordingly.

To approve the presented feature, we used CNN architecture also for this problem. The architecture of the CNN is presented in Figure 7.

Input layer includes

- (i) Seven reduced features of MCI, AD, and HC
- (ii) Input matrix 4D [  $7 \times 1 \times 1 \times 600$  ]

The hidden layers include

- (i) 1D convolution layer
- (ii) Rectified linear unit layer

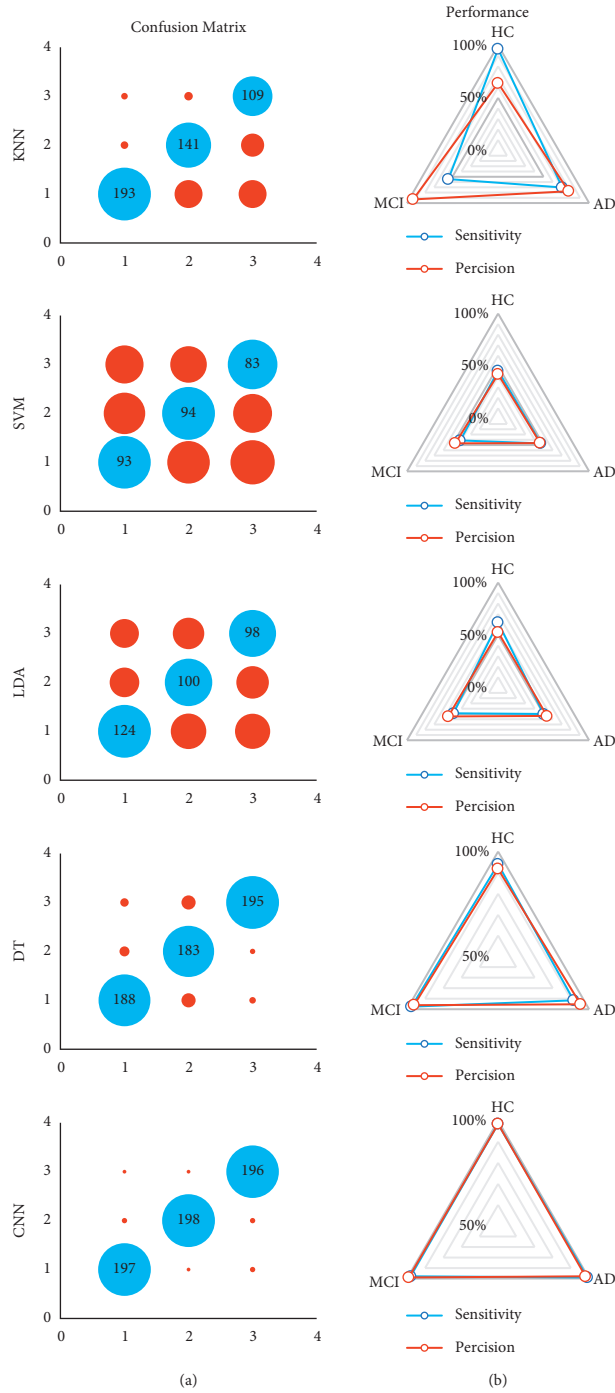


FIGURE 6: The confusion matrixes (a) and the performance plots (b) of the machine learning methods based on the WTD-PSD. The labels 1, 2, and 3 display the HC, AD, and MCI.

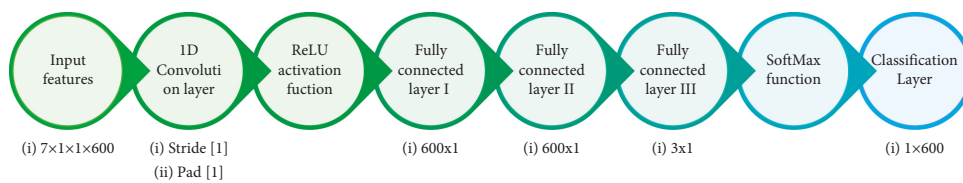


FIGURE 7: The CNN architecture for classification Alzheimer disease based on the WTD-PSD.

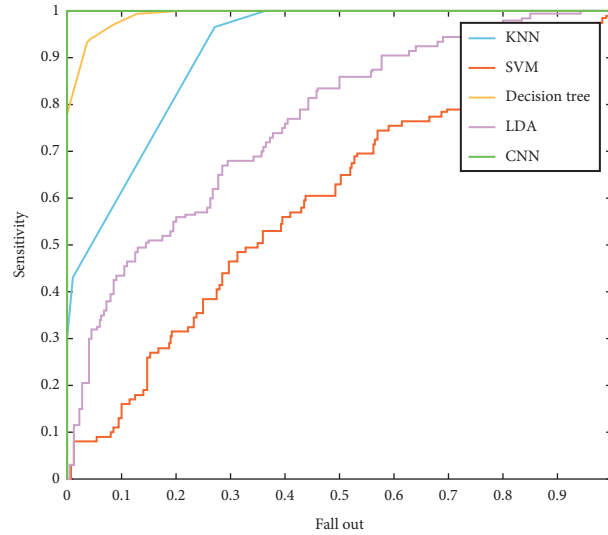


FIGURE 8: The ROC curve for the classification based on the WTD-PSD and machine learning classifiers.

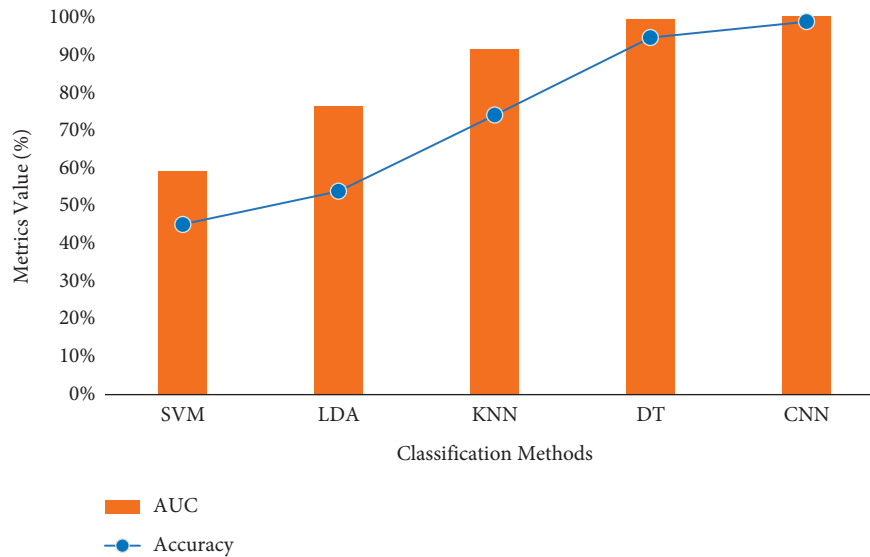


FIGURE 9: The accuracy and AUC value for the presented machine learning methods.

- (iii) Fully connected layer (600)
- (iv) Fully connected layer (600)
- (v) Fully connected Layer (3)
- Output layers include
- (vi) SoftMax layer
- (vii) Classification layer 1D [ $1 \times 600$ ]

Based on the results of the CNN classifier, the sensitivity of the method is 94%, 91.5%, and 97.5% for HC, AD, and MCI, respectively. Moreover, from 200 images for each class, 197, 198, and 196 are detected accordingly. Finally, the precision is 91.7%, 95.3%, and 96.1%, with the same respect. To compare the presented machine learning method for diagnosing Alzheimer's disease, the ROC is depicted in Figure 8. The horizontal axis of

the ROC curve represents the rate of the false-positive index depending on the HC class. The genuine positive rate is shown by the vertical axis. The best classifier has the highest rate of true positives and the lowest number of false positives. Based on the results, the CNN and DT method shows the two best classifiers for the presented features. Moreover, the area under the curve (AUC) value is an index to compare the classifiers. The AUC and the accuracy of the machine learning classifiers are presented in Figure 9. Centered on the results, the accuracy of SVM, LDA, KNN, DT, and CNN is 45%, 53.70%, 73.80%, 94.33%, and 98.50%, respectively. Based on this chart, the CNN architecture with the highest accuracy and AUC is the more accurate and compatible method for diagnosing Alzheimer's disease using the WTD-PSD. Moreover, DT is the second method with a higher AUC.

## 5. Discussion

Since each data sample in ML should be defined as a feature vector, several researches have recommended extracting various features from MRI scans and then categorizing the vectors generated as a consequence of this process. Image preprocessing, on the other hand, is necessary to increase the quality of the recovered feature vectors because of registration mistakes and noise in the image. It is necessary to have domain knowledge in order to derive discriminative qualities. Discrete wavelet is employed as the input image in this study, and it has a two-dimensional representation. The subbanded wavelet coefficients are modeled using the Time-Dependent Power Spectrum Descriptors model, which is implemented in MATLAB. Each of the attributes of the TD-PSD model is represented by one of the leading property vectors. The collected characteristics are utilized in an autonomous manner to construct AD classifications, which are based on classification algorithms. On the basis of the findings, the accuracy of SVM, LDA, KNN, DT, and CNN are correspondingly 45 percent, 53.70 percent, 73.80 percent, 94.33 percent, and 98.50 percent. SVM is the most accurate of the five models. According to this figure, the CNN architecture with the highest accuracy and AUC is the most accurate and compatible technique for diagnosing Alzheimer's disease when utilizing the WTD-PSD than the other two methods. Furthermore, DT is the second most accurate approach with a larger AUC.

## 6. Conclusion

Many studies have advised extracting numerous features from MRI scans and then categorizing the resulting vectors since each data sample in ML should be described as a feature vector. However, image preprocessing is required to improve the quality of the recovered feature vectors due to registration errors and noise. For extracting discriminative characteristics, domain knowledge is required. The Two-Dimensional Discrete Wavelet is used as the input image in this work. The Time-Dependent Power Spectrum Descriptors model is used to model the subbanded wavelet coefficients. The leading property vector is made up of the characteristics of the TD-PSD model. Based on classification algorithms, the extracted features are applied independently to present AD classifications. The classification is used to determine the kind of tumor. We extracted wavelet subband features from three sets of MCI, AD, and HC data using the TD-PSD method. According to the KNN approach, images 193, 141, and 109 are correctly detected from 200 input HC, AD, and MCI images. According to the findings, the KNN's sensitivity for identifying AD in HC patients is adequate. According to the findings, the SVM and LDA approaches yielded a poor outcome for diagnosing AD. The DT findings, on the other hand, demonstrate that the method's sensitivity is 94 percent, 91.5 percent, and 97.5 percent, respectively. It indicates that for this issue, the WTD-PSD is compatible with the DT technique. In other words, MRI images from HC, AD, and MCI are observed in 188, 183, and 195, respectively. Furthermore, the method's precision for HC, AD, and MCI is 91.70 percent, 95.30 percent, and 96.10

percent, respectively. According to the CNN classifier's findings, the method's sensitivity for HC, AD, and MCI is 94 percent, 91.5 percent, and 97.5 percent, respectively. Furthermore, out of 200 images, 197, 198, and 196 are recognized for each class. Eventually, 91.7 percent, 95.3 percent, and 96.1 percent precision are achieved. The CNN architecture with the greatest accuracy and AUC is the more accurate and compatible technique for diagnosing AD utilizing the WTD-PSD, according to this figure. DT is also the second approach with the highest AUC.

## Data Availability

Data are available and can be provided upon direct request to the corresponding author at ali.taghavi.eng@iauctb.ac.ir.

## Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

## References

- [1] M. Rumman, A. N. Tasneem, S. Farzana, M. I. Pavel, and M. A. Alam, "Early detection of Parkinson's disease using image processing and artificial neural network," in *Proceedings of the 2018 Joint 7th International Conference on Informatics, Electronics and Vision and 2nd International Conference on Imaging, Vision and Pattern Recognition, ICIEV-IVPR*, pp. 256–261, IEEE, Kitakyushu, Japan, 25 June 2018.
- [2] K. K. Gulhare, S. P. Shukla, S. P. Shukla, and L. K. Sharma, "Deep neural network classification method to Alzheimer's disease detection," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 7, no. 6, pp. 1–4, 2017.
- [3] A. Nordberg, J. O. Rinne, A. Kadir, and B. Långström, "The use of PET in Alzheimer disease," *Nature Reviews Neurology*, vol. 6, no. 2, pp. 78–87, 2010.
- [4] M. L. Giger, "Machine learning in medical imaging," *Journal of the American College of Radiology*, vol. 15, no. 3, pp. 512–520, 2018.
- [5] Q. Wang, Y. Shi, and D. Shen, "Machine learning in medical imaging," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 4, pp. 1361–1362, 2019.
- [6] F. Li and M. Liu, "Alzheimer's disease diagnosis based on multiple cluster dense convolutional networks," *Computerized Medical Imaging and Graphics*, vol. 70, pp. 101–110, 2018.
- [7] H.-I. Suk, S. W. Lee, S.-W. Lee, and D. Shen, "Latent feature representation with stacked auto-encoder for AD/MCI diagnosis," *Brain Structure and Function*, vol. 220, no. 2, pp. 841–859, 2015.
- [8] M. Liu, D. Zhang, E. Adeli, and D. Shen, "Inherent structure-based multiview learning with multitemplate feature representation for alzheimer's disease diagnosis," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 7, pp. 1473–1482, 2016.
- [9] S. Ji, W. Xu, M. Yang, and K. Yu, "3D Convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2013.
- [10] E. Sadeghipour and N. Sahragard, "Face recognition based on improved SIFT algorithm," *International Journal of Advanced*

- Computer Science and Applications*, vol. 7, no. 1, pp. 547–551, 2016.
- [11] U. R. Acharya, S. L. Fernandes, J. E. WeiKoh et al., “Automated detection of alzheimer’s disease using brain MRI images- A study with various feature extraction techniques,” *Journal of Medical Systems*, vol. 43, no. 9, pp. 1–14, 2019.
  - [12] E. Sadeghipour, N. Sahragard, M. R. Sayebani, and R. Mahdizadeh, “Breast cancer detection based on a hybrid approach of firefly algorithm and intelligent systems,” *Indian Journal of Fundamental and Applied Life Sciences*, vol. 5, p. S1, 2014.
  - [13] E. Sadeghipoor and M. Moradisabzevar, “Screening children with autism by developing smart toy cars,” *International Journal of Research in Engineering and Science*, vol. 2, no. 10, pp. 23–30, 2014.
  - [14] G. Zhou, R. Deng, X. Zhou et al., “Gaussian inflection point selection for LiDAR hidden echo signal decomposition,” *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.
  - [15] Y. Yan, L. Feng, M. Shi, C. Cui, and Y. Liu, “Effect of plasma-activated water on the structure and in vitro digestibility of waxy and normal maize starches during heat-moisture treatment,” *Food Chemistry*, vol. 306, Article ID 125589, 2020.
  - [16] E. Elham and H.-B. Yun, “Attention-based multi-scale convolutional neural network (A + MCNN) for multi-class classification in road images,” *Sensors*, vol. 21, no. 15, p. 5137, 2021.
  - [17] E. Sadeghipour and A. Hatam, “Designing an intelligent system for diagnosing diabetes with the help of the XCSLA system,” vol. 14, no. 1, pp. 24–32, 2015.
  - [18] S. Hassantabar, M. Ahmadi, and A. Sharifi, “Diagnosis and detection of infected tissue of COVID-19 patients based on lung X-ray image using convolutional neural network approaches,” *Chaos, Solitons & Fractals*, vol. 140, Article ID 110170, 2020.
  - [19] E. Sadeghipour and A. Hatam, “An expert clinical system for diagnosing obstructive sleep apnea with help from the XCSR classifier,” vol. 14, no. 1, pp. 33–41, 2015.
  - [20] M. Q. H. Abadi, S. Rahmati, A. Sharifi, and M. Ahmadi, “HSSAGA: designation and scheduling of nurses for taking care of COVID-19 patients using novel method of hybrid salp swarm algorithm and genetic algorithm,” *Applied Soft Computing*, vol. 108, Article ID 107449, 2021.
  - [21] M. Odusami, R. Maskeliūnas, R. Damaševičius, and T. Krilavičius, “Analysis of features of alzheimer’s disease: detection of early stage from functional brain changes in magnetic resonance images using a finetuned ResNet18 network,” *Diagnostics*, vol. 11, no. 6, p. 1071, 2021.
  - [22] A. Sharifi, M. Ahmadi, M. A. Mejni, S. Jafarzadeh Ghouschi, and Y. Pourasad, “Experimental and numerical diagnosis of fatigue foot using convolutional neural network,” *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 24, no. 16, pp. 1828–1840, 2021.
  - [23] S. Dorosti, S. Jafarzadeh Ghouschi, E. Sobhrakhshankhah, M. Ahmadi, and A. Sharifi, “Application of gene expression programming and sensitivity analyses in analyzing effective parameters in gastric cancer tumor size and location,” *Soft Computing*, vol. 24, no. 13, pp. 9943–9964, 2020.
  - [24] M. Ahmadi, A. Taghavirashidizadeh, D. Javaheri, A. Masoumian, S. J. Ghouschi, and Y. Pourasad, “DQRE-SCnet: a novel hybrid approach for selecting users in federated learning with deep-Q-reinforcement learning based on spectral clustering,” *Journal of King Saud University-Computer and Information Sciences*, 2021.
  - [25] A. Ala, F. E. Alsaadi, M. Ahmadi, and S. Mirjalili, “Optimization of an appointment scheduling problem for healthcare systems based on the quality of fairness service using whale optimization algorithm and NSGA-II,” *Scientific Reports*, vol. 11, no. 1, pp. 19816–19819, 2021.
  - [26] N. M. Yazdani, M. S. Panahi, and E. S. Poor, “Intelligent detection of intrusion into databases using extended classifier system,” *International Journal of Electrical and Computer Engineering*, vol. 3, no. 5, 2013.
  - [27] A. Lauraitis, R. Maskeliūnas, R. Damasevicius, D. Polap, and M. Wozniak, “A smartphone application for automated decision support in cognitive task based evaluation of central nervous system motor disorders,” *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 5, pp. 1865–1876, 2019.
  - [28] E. Sadeghipour, A. Hatam, and F. Hosseinzadeh, “Designing an intelligent system for diagnosing diabetes with the help of the XCSLA system,” *Journal of Mathematics and Computer Science*, vol. 14, no. 01, pp. 24–32, 2015.
  - [29] F. Farahanipad, M. Rezaei, A. Dillhoff, F. Kamangar, and V. Athitsos, “A pipeline for hand 2-D keypoint localization using unpaired image to image translation,” in *Proceedings of the The 14th Pervasive Technologies Related to Assistive Environments Conference*, pp. 226–233, ACM, Corfu Greece, 29 June 2021.
  - [30] M. Shi, F. Wang, P. Lan et al., “Effect of ultrasonic intensity on structure and properties of wheat starch-mono-glyceride complex and its influence on quality of norther-style Chinese steamed bread,” *Lebensmittel-Wissenschaft & Technologie*, vol. 138, Article ID 110677, 2021.
  - [31] E. Sadeghipour, A. Hatam, and F. Hosseinzadeh, “An expert clinical system for diagnosing obstructive sleep apnea with help from the xcsr classifier,” *The Journal of Mathematics and Computer Science*, vol. 14, no. 01, pp. 33–41, 2015.
  - [32] M. Rezaei, F. Farahanipad, A. Dillhoff, R. Elmasri, and V. Athitsos, “Weakly-supervised hand part segmentation from depth images,” in *Proceedings of the The 14th Pervasive Technologies Related to Assistive Environments Conference*, pp. 218–225, ACM, Corfu Greece, 29 June 2021.
  - [33] L. Yue, X. Gong, J. Li, H. Ji, M. Li, and A. K. Nandi, “Hierarchical feature extraction for early alzheimer’s disease diagnosis,” *IEEE Access*, vol. 7, pp. 93752–93760, 2019.
  - [34] S. Yang, J. M. S. Bornot, R. B. Fernandez, F. Deravi, K. Wong-Lin, and G. Prasad, “Integrated space-frequency-time domain feature extraction for MEG-based Alzheimer’s disease classification,” *Brain Informatics*, vol. 8, no. 1, pp. 1–11, 2021.
  - [35] R. Hedayati, M. Khedmati, and M. Taghipour-Gorjikolaie, “Deep feature extraction method based on ensemble of convolutional auto encoders: application to Alzheimer’s disease diagnosis,” *Biomedical Signal Processing and Control*, vol. 66, Article ID 102397, 2021.
  - [36] G. Biagetti, P. Crippa, L. Falaschetti, S. Luzzi, and C. Turchetti, “Classification of alzheimer’s disease from EEG signal using robust-PCA feature extraction,” *Procedia Computer Science*, vol. 192, pp. 3114–3122, 2021.
  - [37] Y. Chen and Y. Xia, “Iterative sparse and deep learning for accurate diagnosis of Alzheimer’s disease,” *Pattern Recognition*, vol. 116, Article ID 107944, 2021.
  - [38] M. Ahmadi, A. Sharifi, M. Jafarian Fard, and N. Soleimani, “Detection of brain lesion location in MRI images using convolutional neural network and robust PCA,” *International Journal of Neuroscience*, pp. 1–12, 2021.
  - [39] M. Amini, M. Pedram, A. Moradi, and M. Ouchani, “Diagnosis of alzheimer’s disease severity with fMRI images using robust multitask feature extraction method and convolutional

- neural network (CNN),” *Computational and Mathematical Methods in Medicine*, vol. 2021, pp. 1–15, Article ID 5514839, 2021.
- [40] R. R. Janghel and Y. K. Rathore, “Deep convolution neural network based system for early diagnosis of alzheimer’s disease,” *Irbm*, vol. 42, no. 4, pp. 258–267, 2021.
- [41] M. Ahmadi, A. Sharifi, S. Hassantabar, and S. Enayati, “QAIS-DSNN: tumor area segmentation of MRI image with optimized quantum matched-filter technique and deep spiking neural network,” *BioMed Research International*, vol. 2021, Article ID 6653879, 16 pages, 2021.
- [42] H. Parmar, B. Nutter, R. Long, S. Antani, and S. Mitra, “Spatiotemporal feature extraction and classification of Alzheimer’s disease using deep learning 3D-CNN for fMRI data,” *Journal of Medical Imaging*, vol. 7, no. 05, Article ID 56001, 2020.
- [43] M. Ahmadi, F. Dashti Ahangar, N. Astaraki, M. Abbasi, and B. Babaei, “FWNNNet: presentation of a new classifier of brain tumor diagnosis based on fuzzy logic and the wavelet-based neural network using machine-learning methods,” *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 8542637, 13 pages, 2021.
- [44] Y. Li, J. Jiang, J. Lu, J. Jiang, H. Zhang, and C. Zuo, “Radiomics: a novel feature extraction method for brain neuron degeneration disease using 18F-FDG PET imaging and its implementation for Alzheimer’s disease and mild cognitive impairment,” *Therapeutic Advances in Neurological Disorders*, vol. 12, 2019.
- [45] G. Fiscon, E. Weitschek, A. Cialini et al., “Combining EEG signal processing with supervised methods for Alzheimer’s patients classification,” *BMC Medical Informatics and Decision Making*, vol. 18, no. 1, pp. 1–10, 2018.
- [46] I. R. R. Silva, G. S. L. Silva, R. G. De Souza, W. P. Dos Santos, and R. A. de A Fagundes, “Model based on deep feature extraction for diagnosis of alzheimer’s disease,” *2019 International Joint Conference on Neural Networks (IJCNN)*, vol. 2019, pp. 1–7, 2019.
- [47] Y. Zhang, S. Wang, Y. Huo, L. Wu, and A. Liu, “Feature extraction of brain MRI by stationary wavelet transform and its applications,” *Journal of Biological Systems*, vol. 18, no. SPEC. ISSUE 1, pp. 115–132, 2010.
- [48] C. eetha and D. Pugazhenthii, “Classification of alzheimer’s disease subjects from MRI using fuzzy neural network with feature extraction using discrete wavelet transform,” *Bio-medical Research*, vol. 2018, pp. S14–S21, 2018.
- [49] W. Badawy, M. Weeks, G. Zhang, M. Talley, and M. A. Bayoumi, “MRI data compression using a 3-D discrete wavelet transform,” *IEEE Engineering in Medicine and Biology Magazine*, vol. 21, no. 4, pp. 95–103, 2002.
- [50] J. L. Whitwell, K. A. Josephs, M. E. Murray et al., “MRI correlates of neurofibrillary tangle pathology at autopsy: a voxel-based morphometry study,” *Neurology*, vol. 71, no. 10, pp. 743–749, 2008.
- [51] P. Vemuri, J. L. Whitwell, K. Kantarci et al., “Antemortem MRI based STructural Abnormality iNDex (STAND)-scores correlate with postmortem Braak neurofibrillary tangle stage,” *NeuroImage*, vol. 42, no. 2, pp. 559–567, 2008.
- [52] B. Saeed, *Alzheimers\_Brain\_MRI*, Kaggle, San Francisco, US, Dec. 2019, <https://www.kaggle.com/basheersaeed/alzheimers-brain-mri>.

## Research Article

# PSOWNNs-CNN: A Computational Radiology for Breast Cancer Diagnosis Improvement Based on Image Processing Using Machine Learning Methods

Ashkan Nomani,<sup>1</sup> Yasaman Ansari,<sup>2</sup> Mohammad Hossein Nasirpour,<sup>3</sup> Armin Masoumian,<sup>4</sup> Ehsan Sadeghi Pour,<sup>5</sup> and Amin Valizadeh <sup>6</sup>

<sup>1</sup>Department of Electrical and Computer Engineering, Drexel University, Philadelphia, PA, USA

<sup>2</sup>Department of Computer Engineering, Tehran North Branch, Islamic Azad University, Tehran, Iran

<sup>3</sup>Department of Medical Genetics, Institute of Medical Biotechnology, National Institute of Genetic Engineering and Biotechnology (NIGEB), Tehran, Iran

<sup>4</sup>Department of Computer Engineering and Mathematics, Universitat Rovira I Virgili, Tarragona, Spain

<sup>5</sup>Department of Electrical and Computer Engineering, Kashan Branch, Islamic Azad University, Kashan 8715998151, Iran

<sup>6</sup>Department of Mechanical Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

Correspondence should be addressed to Amin Valizadeh; [amin.valizadeh@mail.um.ac.ir](mailto:amin.valizadeh@mail.um.ac.ir)

Received 23 February 2022; Accepted 29 March 2022; Published 11 May 2022

Academic Editor: Mohamed Abdelaziz

Copyright © 2022 Ashkan Nomani et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Early diagnosis of breast cancer is an important component of breast cancer therapy. A variety of diagnostic platforms can provide valuable information regarding breast cancer patients, including image-based diagnostic techniques. However, breast abnormalities are not always easy to identify. Mammography, ultrasound, and thermography are some of the technologies developed to detect breast cancer. Using image processing and artificial intelligence techniques, the computer enables radiologists to identify chest problems more accurately. The purpose of this article was to review various approaches to detecting breast cancer using artificial intelligence and image processing. The authors present an innovative approach for identifying breast cancer using machine learning methods. Compared to current approaches, such as CNN, our particle swarm optimized wavelet neural network (PSOWNN) method appears to be relatively superior. The use of machine learning methods is clearly beneficial in terms of improved performance, efficiency, and quality of images, which are crucial to the most innovative medical applications. According to a comparison of the process's 905 images to those of other illnesses, 98.6% of the disorders are correctly identified. In summary, PSOWNNs, therefore, have a specificity of 98.8%. Furthermore, PSOWNNs have a precision of 98.6%, which means that, despite the high number of women diagnosed with breast cancer, only 830 (95.2%) are diagnosed. In other words, 95.2% of images are correctly classified. PSOWNNs are more accurate than other machine learning algorithms, SVM, KNN, and CNN.

## 1. Introduction

In the world, breast cancer is one of the leading health problems for women. Breast cancer comes in second place to lung cancer in terms of incidence. According to studies, one out of every nine women will be diagnosed with breast cancer. Approximately 2,088,849 cases of breast cancer were diagnosed globally in 2018 (11.6 percent of all cancer diagnoses) [1, 2]. Breast cancer occurs when there is an

overdevelopment of cells in the breast, resulting in lumps or tumors. Malignant tumors tend to penetrate their surroundings more readily and are considered to be cancerous. Benign tumors are less likely to do this [3]. The masses are usually left untreated if they do not cause discomfort to the breast or spread to neighboring tissues. Many types of benign lumps can be found in breasts and prostates, including cysts, fibroadenomas, phyllodes tumors, atypical hyperplasias, and fat necrosis. Tumors can be malignant or invasive.



In the absence of early diagnosis and treatment, these lesions spread and damage the surrounding breast tissues, leading to metastatic breast cancer [4, 5]. Metastatic breast cancer occurs when breast tumor cells spread to other organs, such as the liver, brain, bones, or lungs, through the bloodstream or lymphatic system [6]. Breast tissue is mostly made up of glandular (milk-producing) and fat tissues, as well as lobes and ducts. There are numerous types of breast cancer. Ductal and lobular carcinomas are the two most common types of invasive breast cancer [7–9]. In addition to redness, swelling, scaling, and underarm lumps, breast cancer survivors also notice irritation to the skin, fluid leakage, and distorted breasts. The five stages of breast cancer (stages 0 through IV) range from noninvasive malignancy to aggressive breast cancer. There are over 90,000 new cases of these illnesses every year in Asia, and 40,000 people die from them. In part, the growing death rate is due to a lack of knowledge, low education levels, and widespread poverty in diagnosis or consultation with physicians. It may be possible to significantly increase the chance of survival and find more effective treatment options if this condition is diagnosed early. Mammography can reduce mortality by one-third for women over the age of 50 [10, 11].

Because breast cancer cannot be prevented, many manual and image-based exams are useful for identifying and diagnosing it. For early detection of this disease, women are advised to perform a self-exam to become aware of the frequency of bizarre breast anomalies. Breast cancer screenings use a variety of imaging techniques, including X-ray mammography, ultrasound MRI, thermography, and CT scans [12, 13]. Researchers can use these images to examine several breast cancer-related issues. Breast cancer may appear on mammograms as microcalcifications, masses, and architectural deformities, but WSI can also detect abnormalities in the nucleus, epithelium, tubules, stroma, and mitotic activity in breast tissue [14]. In the absence of a tumor, architectural distortion is the hardest abnormality to detect on mammography. Medical breast imaging, such as mammography, is often interpreted differently by expert radiologists. Breast Imaging Reporting and Dated System (BIRADS) was developed by the American College of Radiology to deal with this conflict and radiologists' subjectivity during interpretation and features of breast mammograms, ultrasounds, and magnetic resonance imaging (MRI). Researchers have pioneered the development of artificial neural networks (ANNs) for the detection of breast cancer in recent years. An important aspect of this invention is calculating how many aspects of a diagnostic procedure can be positively affected [15–17].

Additionally, automated detection of breast cancer can mimic the unique behavior of the human brain, making it more effective than manual methods [18, 19]. An ANN cancer detection system mimics the functions of the human brain by approximating and resolving nonlinear and difficult issues, which can be perceived as a mathematical representations-inspired learning process. Further, the predictive accuracy of ANN-based cancer diagnosis is better than that of classic statistical detection approaches due to the latter's reliance on parameter optimization [12, 20]. An ANN-based

cancer detection method's performance is also affected by (a) feature selection, (b) learning algorithms and their rates, (c) hidden layer count, (d) multiple nodes in a hidden layer, and (e) initial weights for the factors considered during optimization. When developing ANN-based breast cancer detection systems, feature selection is perhaps the most important factor to consider. ANN-based breast cancer detection techniques rely heavily on feature subsets [21, 22]. Additionally, the input feature subset and the design elements in the ANN-inspired breast cancer diagnostic have a reciprocal relationship. Therefore, the ANN-based process of breast cancer diagnosis must be optimized in terms of feature subset and design parameters [23].

The aim of this research is to reduce uncertainty in order to improve accuracy. Throughout history, uncertainty has always played a role in decision-making, and this is evident by the lack of clarity in the problems. There are times when it is impossible to predict all the parameters of a system, resulting in an incorrect choice. The remainder of this article is organized as follows. The purpose of an artificial neural network is to take in input in the form of a radiological discovery and to generate output in the form of a biopsy. A neural network can be used to identify and predict the risk of breast cancer in masses. In mammography, machine learning methods are used to identify abnormalities by classifying suspicious areas. In the Conclusion, a full assessment of the findings will be presented.

## 2. Related Work

Various deep-learning algorithms have been successfully used to build automated digital models in a variety of applications [24–27]. Using the discrete wavelet transform (DWT) and back-propagation neural networks (BPNN), Beura et al. developed a CAD model based on GLCM features and a BPNN classifier [28]. A KNN classifier was used in conjunction with DWT and GLCM features to develop a CAD model. Based on principal component analysis (PCA) and a support vector machine (SVM) classifier, Liu et al. provided a model that reduced DWT features. DWT and SVM-based CAD models were suggested by Basheer et al. [29]. Linear Discriminant Analysis (LDA) is used in a KNN classifier to extract salient features from a discrete curvelet transform (DCT) model described by [30]. Using lifting wavelet transform features and an extreme learning machine (ELM), Muduli et al. developed a moth flame optimization algorithm to build a hybrid classifier [31]. It produces better classification results with fewer features. Based on support vector machines (SVMs) in particle swarm optimization (PSO), Khan et al. [32] developed an optimized Gabor filter bank CAD model to extract important features and then improve accuracy by using SVM classifiers. The use of ultrasound images for breast cancer classification has also been introduced using machine-learning-based models. A neural network is employed to classify a feature-based model based on autocorrelation coefficients, proposed by Xiao et al. [33]. According to Liu et al. [34], repairing damaged fonts based on style is a better method of repairing damaged fonts.

Researchers have found that the font content provided by the research-based CGAN network repair style is comparable to the right font content. Zhou et al. [35] described an efficient blind quality assessment approach for SCIs and NSIs that is based on a dictionary of learnt local and global quality criteria. Li et al. [36] created an artificial intelligence technique that is used for data-enhanced encryption at the IoT's endpoints and intermediate nodes. The technique presented in this article is an AI approach for encrypting data at the endpoints and intermediate nodes of autonomous IoT networks. Yang et al. [37] presented a temporal model for page dissemination in order to reduce the disparity between prediction data from current models and actual code dissemination data. In a study by Eslami et al. [38], attention-based multiscale convolutional neural networks (A+MCNN) were used to efficiently separate distress items from non-distress items in pavement photos. Liao et al. [39] developed an enhanced faster regions with CNN features (R-CNN) technique for semi-supervised SAR target identification that includes a decoding module and a domain-adaptation module named FDDA. Liu et al. [40] developed self-supervised CycleGAN in order to achieve perception consistency in ultrasound images. Sharifi et al. [41] shown how to diagnose tired and untired feet using digital footprint images. According to Zuo et al. [42], deep-learning technologies have improved optical metrology in recent years. He et al. [43] introduced a number of feature selection techniques for reducing the dimensionality of data. Ahmadi et al. [44] developed a new classifier based on wavelet transformation and fuzzy logic. The ROC curve findings show that the given layer is able to accurately segment brain tumors. To predict m6A from mRNA sequences, Zou et al. [45] used word embedding and deep neural networks. Jin et al. [46] developed word embedding and deep neural networks for m6A prediction from mRNA sequences. Yang et al. [47] sought to elucidate the mechanism behind the movement of soy husk polysaccharide (SHP) in the mucus layer triggered by  $\text{Na}^+/\text{Ca}^{2+}$ . The findings indicated that  $\text{Na}^+$  had minimal influence on the viscosity of polysaccharides, but  $\text{Ca}^{2+}$  enhanced it. Using a speckle-emphasis-based feature combined with an SVM classifier, Chang et al. [48] suggested a multifeature extraction model that provides the best results. A model that incorporates curvelet, shearlet, contourlet, wavelet, and gray-level cooccurrence matrix (GLCM) features has been proposed by Zhou et al. [49]. For optimal breast cancer detection, Liu et al. [50] proposed an interval-uncertainty-based strategy. Indeterminacy was accounted for using interval analysis. Regardless of the imaging system's alterations, the approach is guaranteed to provide acceptable results. To develop an interval-based Laplacian of Gaussian filter which can be used to simulate intensity uncertainties, the goal was to develop an interval-based Laplacian of Gaussian filter. To demonstrate the method's effectiveness, final findings were applied to the MIAS database and compared with several established methodologies.

A CNN-based method of detecting breast cancer was proposed by Zuluaga et al. [51]. This method was enhanced by BreastNet. Prior to including the image data into the model, the

expansion approach was used to establish the image data. An accurate classification system was developed using hyper-column methodology. To demonstrate the recommended system's increased accuracy, the findings were compared to those of several recent approaches. In histopathology images, Carvalho et al. [52] employed a different method of detecting breast cancer. Phylogenetic diversity indices were used for the construction of models and the categorization of histopathological breast images by the authors. To test its accuracy, the approach was compared to a variety of other recent methodologies. Mahmood et al.'s [53] unique convolutional neural network (ConvNet) used deep learning to identify breast cancer tissues with dramatically lower human error. For identifying mammographic breast masses, the proposed technique obtained a spectacular training accuracy of 0.98, an AUC of 0.99, high sensitivity of 0.99, and test accuracy of 0.97. According to Zhang et al. [54], different identification and detection methods pose both challenges and opportunities, such as amplification of nucleic acids, optical POCT, electrochemistry, lateral flow assays, microfluidics, enzyme-linked immunosorbent assays, and microarrays. Jiang et al. [55] focused on the surface teeth of the entire crown. Robot-assisted trajectory planning is demonstrated to improve efficiency and alleviate pressure associated with manual tooth preparation within the margin of error. Its practicability and validity are demonstrated. Qin et al. [56] suggested a novel monitoring technique for robotic drilling noise based on focused velocity synchronous linear chirplet transforms. Mobasher et al. [57] reviewed important immunological results in COVID-19 and contemporary reports of autoimmune illnesses related to the condition. According to Ala et al. [58], for solving the appointment scheduling model using a simulation technique, they developed the whale optimization algorithm (WOA), which uses the Pareto archive and the NSGA-II algorithm. An adaptive secondary sampling method based on machine learning for multiphase drive systems is proposed by Liu et al. [59]. Zheng et al. [60] recommended image classification as the research goal for examining how metalearning rapidly acquires knowledge from a limited number of sample photos. In an article, Liu et al. [61] developed an image stitching algorithm based on feature point pair purification. Kaur et al. [62] have used a deep convolutional neural network (DCNN) and fuzzy support vector machines; they have developed two-class and three-class models for breast cancer detection and classification. Mammogram images from DDSM and curated breast imaging subsets DDSM (CBIS-DDSM) were used to create the models. Our system was tested for accuracy, sensitivity, AUC, F1-score, and confusion matrix. For the DCNN and fuzzy SVM, the accuracy of the 3-class model was 81.43 percent. With a 2-layer serial DCNN with fuzzy SVM, the first layer achieved accuracies of 99.61 percent and 100.00 percent, respectively, in binary prediction. Table 1 shows the summary of related work.

To get high-frequency and low-frequency pictures, Li et al. [69] employed a wavelet for multiscale decomposition of the source and fusion images. This article employed a deep convolutional neural network to learn the direct mapping between the high-frequency and low-frequency pictures of the source and fusion images in order to get clearer and more comprehensive fusion images. The experimental

TABLE 1: Summary of related work.

Author	Year	Type	Network	Result	Advantages	Disadvantages
Dong et al. [22]	2022	Breast cancer diagnosis and classification	Random forest and regression tree	The application of machine learning techniques like CART and random forests coupled with geographical methodologies provides a viable alternative for future inequalities studies	(i) Low complexity (ii) High accuracy	(i) Possible overfitting (ii) Used classic feature extraction (iii) Lowest robustness
Guha et al. [19]	2022	Breast cancer risk factors	SEER-Medicare analysis	The incidence of AF in women after a breast cancer diagnosis is much higher. AF is strongly linked to a higher stage of breast cancer upon diagnosis. Women newly diagnosed with breast cancer who develop AF suffer an increased risk of cardiovascular death but not breast-cancer-related death	(i) Ability of risk assessment (ii) Technical assessment	(i) Needs feature extraction (ii) Unable to diagnose the patient
Chamieh et al. [63]	2022	Breast cancer diagnosis using fine-needle aspiration cytology	Begg and Greenes method	Irrespective of the recommended technique, the FNAC test's specificity was always greater than its sensitivity. For all approaches, the probability ratios were positive. Both positive and negative yields were high, demonstrating the test's exact discriminating qualities.	(i) Technical assessment method (ii) Low complexity	(i) Unable to diagnose illness type (ii) Limited dataset
Thangarajan et al. [64]	2022	Breast cancer biomarkers validated in plasma	BC gene expression profiling	Methylation status of SOSTDC1, DACT2, and WIF1 can distinguish BC from benign and control conditions with 100% sensitivity and 91% specificity. Therefore, SOSTDC1, DACT2, and WIF1 may be used as a supplemental diagnostic tool to distinguish noninvasive and invasive breast cancer from benign breast conditions and healthy individuals	(i) Using biomarkers instead of mathematical features	(i) Lower sensitivity (ii) Unable to diagnose illness type
Chakravarthy et al. [65]	2022	Diagnosis of breast cancer	Multideep CNN	By fuzzing deep features for both datasets (97.93 percent for MIAS and 96.646 percent for INbreast), we achieved the highest classification accuracy among state-of-the-art frameworks. When the PCA was applied to combined deep features, classification performance did not improve, but execution time was shorter, resulting in a lower computing cost	(i) Low complexity (ii) High accuracy (iii) Technical assessment method	(i) Possible overfitting (ii) Needs feature extraction (iii) Lower sensitivity (iv) Lowest robustness

TABLE 1: Continued.

Wang et al. [66]	2022	Metastasis of breast cancer axillary lymph nodes forecasting	CNN	The T2WI sequence outperformed the other three sequences in the testing set, with accuracy and AUC of 0.933/0.989. In comparison with T1WI, which has accuracy and AUC of 0.691/0.806, the increase is substantial	(i) Ability of risk assessment (ii) Technical assessment	(i) Unable to diagnose the patient (ii) Used classic feature extraction (iii) Limited dataset (iv) Lower sensitivity
Melekoodappattu et al. [67]	2022	Breast cancer detection	CNN and texture feature-based approach	Using our ensemble method, we measured 97.8% specificity and 98.6% accuracy for MIAS and 98.3% and 97.9% for DDSM. Experimental data indicate that the combination strategy enhances measurement metrics independently for each phase	(i) Low complexity (ii) High accuracy	(i) Unable to diagnose illness type
Gonçalves et al. [68]	2022	Breast cancer detection	CNNs	VGG-16 produced F1-scores greater than 0.90 for all three networks, an increase from 0.66 to 0.92. Furthermore, compared to earlier research, we were able to improve the F1-score of ResNet-50 from 0.83 to 0.90	(i) Comparative study (ii) Used high-rank methods	(i) Unable to diagnose illness type (ii) High complexity

findings demonstrated that the approach suggested in this research may produce a good fusion picture that is superior in terms of both visual and objective assessments to that produced by certain sophisticated image fusion algorithms. In their study, Zhang et al. [70] used the Gaussian pyramid to improve the basic ORB-oriented approach. Based on the experimental results, we have demonstrated that the pyramid sphere method is invariant, resilient to scale and rotation changes, and has a high degree of registration accuracy. Additionally, the stitching speed is around ten times to that of SIFT. Shan et al. [71] employed a two-dimensional three-dimensional multimode medical image registration technique based on normalized cross-correlation. The results demonstrate that a multiresolution technique enhances registration accuracy and efficiency by compensating for the normalized cross-correlation algorithm's inefficiency. Xu et al. [72] suggested a technique for segmenting and categorizing tongue pictures using an MTL algorithm. The experimental findings demonstrate that our combined strategy outperforms currently available tongue characterisation techniques. Ahmadi et al. [73] used a CNN to segment tumors associated with seven different types of brain disorders: glioma, meningioma, Alzheimer's, Alzheimer's plus, Pick, sarcoma, and Huntington's. Sadeghipour et al. [74] developed a new method, combining the firefly algorithm and an intelligent system, to detect breast cancer. Researchers Zhang et al. [75] explored a way to query clinical pathways in E-healthcare systems while preserving privacy. According to Sadeghipour et al. [76], a new designed system was developed for diagnosing diabetes using the XCSLA system. Ahmadi et al. [77] introduced a

technique called QAIS-DSNN for segmenting and distinguishing brain malignancies from MRI images. The simulation results obtained using the BraTS2018 dataset indicate that the suggested technique is 98.21 percent accurate. Chen et al. [78] developed a model of label limited convolutional factor analysis (LCCFA) that combines factor analysis (FA), convolutional operations, and supervised learning. Our technique outperforms other relevant models in terms of classification accuracy for small samples on multiple benchmark datasets and measured radar high-resolution range profile data. Rezaei et al. [79] created a data-driven technique for segmenting hand parts on depth maps that does not need additional work to get segmentation labels. Sadeghipour et al. [80] created a clinical system for diagnosing obstructive sleep apnea with the XCSR Classifier's assistance. Dorosti et al. [81] developed a generic model to determine the link between several characteristics in a GC tumor's location and size. Abadi et al. [82] suggested a unique hybrid salp swarm algorithm and genetic algorithm (HSSAGA) model for scheduling and designating nurses. The proposed test function algorithm's results indicate that it outperforms state-of-the-art techniques. Zhou and Arandian [83] proposed a computer-aided technique for skin cancer detection. A mix of deep learning and the Wildebeest Herd Optimization Algorithm was used to create the approach. The first characteristics are extracted using an Inception convolutional neural network. Following that, the WHO method was used to choose the relevant characteristics in order to reduce the analysis time complexity. Finally, the entire diagnostic system was compared to other ways in order to determine its efficacy in comparison to the methods

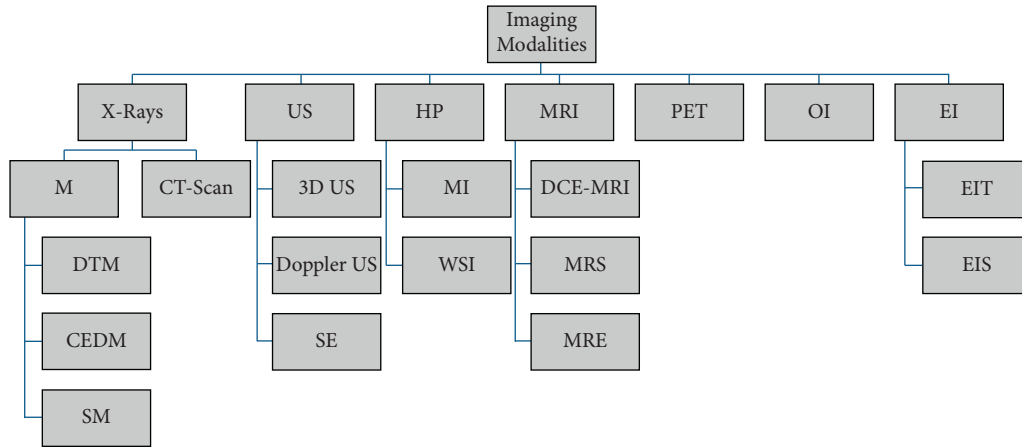


FIGURE 1: Modalities of imaging.

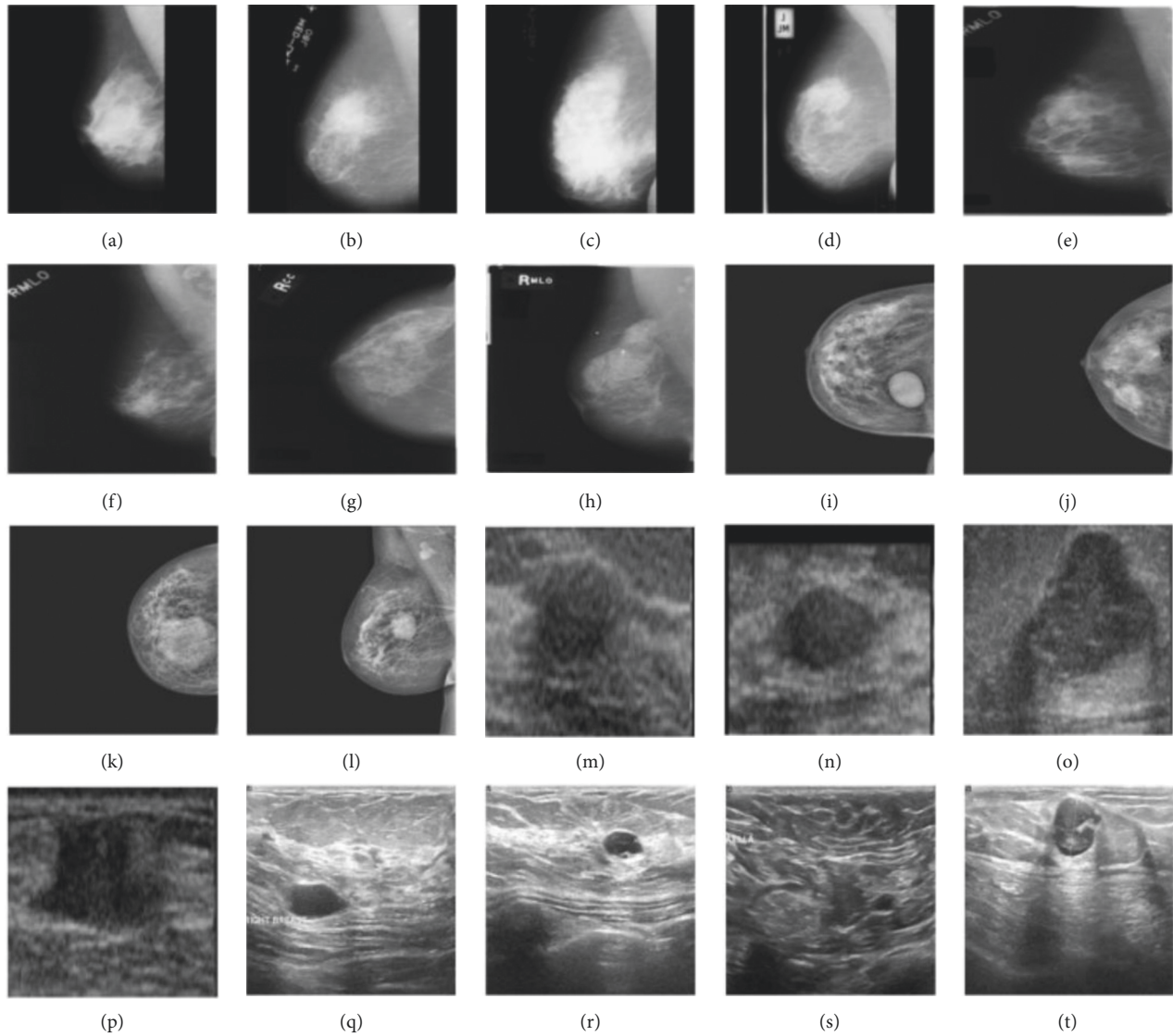


FIGURE 2: Sample benign images from different datasets like MIAS ((a) and (b)), DDSM ((e) and (f)), INbreast ((i) and (j)), BUS-1 ((m) and (n)), and BUS-2 ((q) and (r)). Sample malignant images from different datasets like MIAS ((c) and (d)), DDSM, ((g) and (g)), INbreast ((k) and (l)), BUS-1 ((o) and (p)), and BUS-2 ((s) and (t)).

evaluated. Davoudi et al. [84] examined the effect of statins on the severity of COVID-19 infection in patients who had been taking statins prior to infection. Hassantabar et al. [85] examined the effect of statins on the severity of COVID-19 infection in patients who had been taking statins prior to infection. Yu et al. [86] used differential expression analysis to combine the biological relevance of genes from gene regulatory networks, resulting in weighted differentially expressed genes for breast cancer detection. The binary classifier was capable of making a decent prediction for an unknown sample, and the testing results confirmed the efficacy of our suggested methods. A convolutional neural network based on an artificial fish school technique was suggested by Thilagaraj et al. [87]. The breast cancer image dataset comes from archives of cancer imaging. The breast cancer picture was filtered with the use of a Wiener filter for classification in the preprocessing phase of classification. By determining the number of epochs and training pictures for the deep CNN, the optimization approach assisted in lowering the error rate and increasing performance efficiency.

### 3. Breast Cancer Detection and Diagnosis

Prediction and treatment of breast cancer using computers are largely based on intermediate procedures such as segmenting (identifying breast lesions), identifying features, and finally categorizing areas found into classes. It is possible to detect breast lesions by either defining a suspicious region pixel by pixel in a breast image or by creating a bounding box around the suspicious area. Cancer could be detected by processing whole breast images instead of removing worrisome spots and then categorizing them, which would incur an additional charge. To classify the lesions under investigation, features are extracted from the ROI or the whole image. A classification algorithm (ML or DL) uses these features to classify the samples.

*3.1. Features Learning.* There are many aspects of this work which are depicted through the images. Segmenting and classifying images require knowledge of the most informative and accurate features. A large and complicated set of features are extracted due to the discrepancy between benign and malignant lesions. As a result, selecting the right set of features is crucial, since having too many features can degrade the classifier's performance and increase its complexity. To segment and classify breast lesions, numerous kinds of handmade features, such as texture, size, shape, intensity, and margins, were previously obtained from breast images. [12].

As a result, deep learning has considerably improved the whole feature extraction process, thereby improving the performance of the following stages (e.g., detection and classification). Hence, deep features derived from a convolutional network trained on a large dataset can perform discriminating tasks far better than conventional approaches based on hand-engineered features or typical machine learning methods (see Figure 1).

### 4. Proposed Method

The flow chart of the structure of CNN used can be seen in Section 4.7. Our approach to improving Shafer's hypothesis was a combination of approaches as you can see in the image. Machine learning and neural networks are used to classify and diagnose tumors. For this purpose, CNN deep neural networks are individually trained and tested. This article discusses two strategies for feature extraction. CNN uses deep features for feature extraction. With gray-level cooccurrence matrix features retrieved from the image, an artificial neural network is trained. In the subsequent stages, a classifier is used to determine the probability of each class.

*4.1. Dataset.* This study used Mini-MIAS Screening mammography data as input images. Data would be gathered directly from hospitals and physicians, as well as from public sources. Data would be publicly available. Image resolution is  $256 \times 256$  pixels in PNG format. This is an example of an image. 1824 images are used for analysis and simulation: 80% for training and 20% for validation.

*4.2. Analyzing Outliers and Reducing Dimensions.* To reduce the dimensionality of the data, Principal Component Analysis (PCA) was used. To determine the appropriate number of principal components, several machine learning models were fitted repeatedly to the modified data. In order to evaluate the effects of dimensionality reduction on prediction accuracy, a predefined number of principal component analyses (PCA) were conducted before training all models using the Classifying Learner application in the Statistics and Machine Learning Toolbox. Principal component analysis of the data was used to identify the "base" model. In order to determine the appropriate number of principal components for the modified data, machine learning models were fitted repeatedly. The PCA was performed independently on benign and malignant tumors to remove outliers.

*4.3. ROI Preprocessing.* A variety of undesirable visual features, noise, and artifacts can be seen in the mammography images in the database, resulting in a low-quality image that will inevitably lower the accuracy of classification. As a result of manually cropping each image in the MIAS, we obtained the Region-of-Interest (ROI) encompassed by the suspected anomaly. To crop the image to size, the radiologist provided the center and radius parameters in the dataset (Figure 1). The ROIs that were retrieved are shown in Figure 2. Size has been assigned to the INbreast, BUS-1, and BUS-2 full images due to a lack of ground truth data. Figure 2 shows the MIAS, DDSM, INbreast, BUS-1, and BUS-2 datasets [69].

*4.4. Feature Extraction.* To minimize the number of resources required for an accurate display of a large amount of data, feature extraction is used. When collecting complex data, the number of variables being examined is one of the most significant challenges. Using the instructional example,

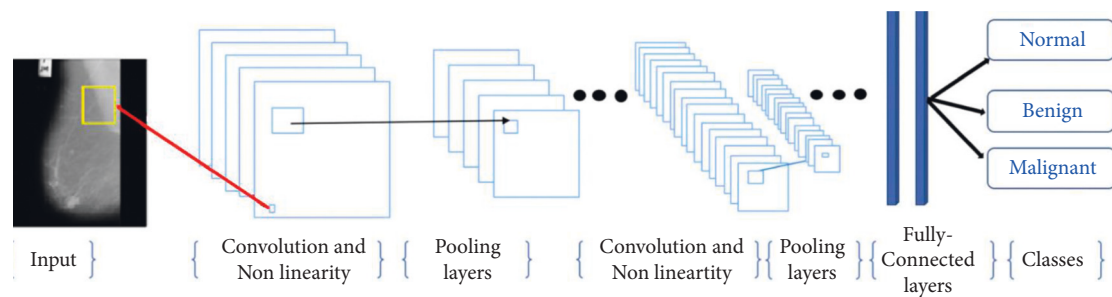


FIGURE 3: A description of a CNN architecture.

a large number of variables requires a large amount of memory and storage. To solve problems requiring high precision, feature extraction is a term that refers to a wide range of methodologies for gathering data. The idea behind image analysis is to design a unique method for representing the fundamental components of an image distinctively. The fractal approach was used to generate gray area vectors for feature vectors. Based on the light intensity of the defined locations relative to someone in the image, statistical analysis is used to create the image features of the confidence interval for the identified chemicals. In each combination of intensity points (pixels), the statistics are affected by the frequency of these points. We extract the feature using the GLCM model in this study. The feature selection technique was used to reduce the dimensions and identify additional critical qualities that might adequately separate the various systems in terms of their capacity to interact with large amounts of input data [48]. The GLCM approach was combined with covariance analysis to extract eigenvalues and reduce the size of the image. The fractal approach requires identical input images, and each image is considered a two-dimensional matrix and a single vector. Images must be grayscale and of a certain resolution. As a result of matrix reshaping, each image becomes a column vector, and each image is extracted from an MN matrix, where N is the number of pixels in each image, and M is the total number of images. It is necessary to compute the average image for each original image in order to establish the normal distribution. The covariance matrix is then calculated as well as the eigenvalues and eigenvectors. Fractal systems use the following mechanism: M is the number of training images,  $F_i$  is the mean of the images, and  $l_i$  represents each image in the  $T_i$  array, beginning with M images, each of which has NN pixels.

#### 4.5. Concept of Convolutional Neural Network (CNN).

Convolutional neural network (CNN) is a significant technique from the deep learning field. A CNN typically includes principal layers of the convolution, the Maxpooling, the fully connected layer, and the other layers executing various features. There are two phases for the preparation of each system, the forward progression and backward progression. Firstly, the data moves from the input layer to the hidden layer and then to the output layer. In the back-propagation algorithm, the input image is doing the feeding process to the network in the first step. Once the output is achieved, the error value is evaluated. This value is then brought back to the network together with updating the

network weight and along with the cost functions diagram (see Figure 3). CNN consists of different types of hidden sublayers as discussed below.

*Convolutional Layer.* It is the principle of the convolution network. This layer's output is represented as a 3D neuron matrix. The CNN network requires multiple kernels in certain layers to transform both the input image and the core function maps. The convolution process has three key advantages:

- (i) The weight-sharing method decreases the number of features in each function diagram.
- (ii) The connection between adjacent pixels is known through a local connection.
- (iii) It induces changes in the position of the target to create equilibrium.

*Activation Functions.* In particular, activation functions are used in neural networks to achieve the desired output. Neural networks can use various activation functions; the most significant of which are Tanh activation functions and Sigmoid. The sigmoid function transforms input data  $(-\infty \rightarrow +\infty)$  to values from 0 to 1. Tanh provides production value 1 to  $-1$  interval. One of the other activation functions is the ReLU function which has been introduced in recent years. ReLU is a function of activation  $g$  extended to all components. It aims to present the network with nonlinear behavior. This functionality contains all pixels in the image and converts all negative points to 0.

*Max pooling:* There are several consequences of the use of Maxpooling in CNN. The use of it helps CNN to define the target with only small modifications to the matrix at first. Second, it helps CNN to recognize features in the huge size of the image. CNN Maxpooling is used to sum up the functions during the sampling process subtraction and so can be gotten into deep steps of CNN. Then we need to begin pooling to get what we have to retain this information. Among the most common forms of pooling are Max and Average.

*Data augmentation:* Preprocessing and data enhancement are some of the most often overlooked issues. However, these tasks are often unnecessary. You should always check whether your task needs to be pre-processed before running any data processing.

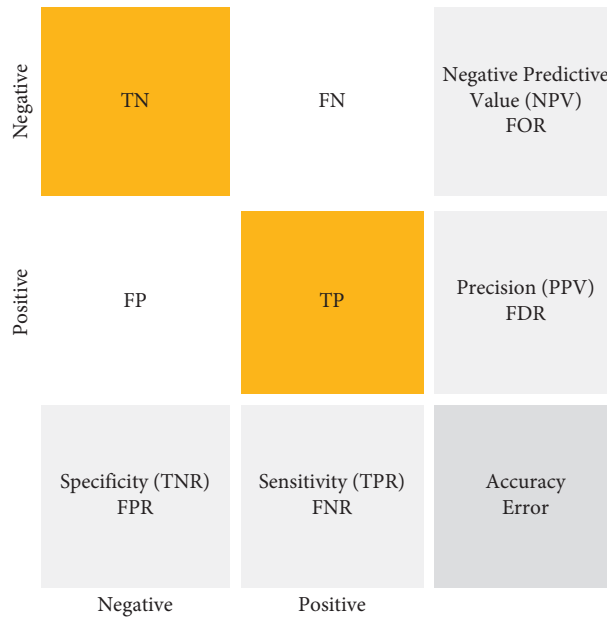


FIGURE 4: The confusion matrix.

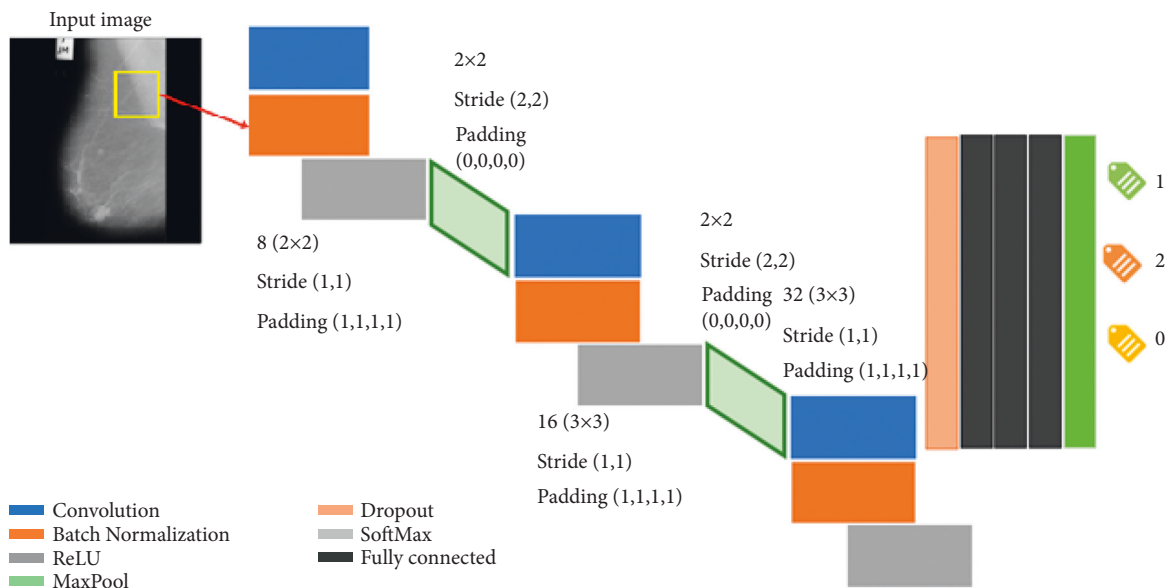


FIGURE 5: The structure of CNN.

4.6. Classifier’s Performance Analysis Metrics. By comparing the true positive rate (TPR) to the false positive rate (FPR), the receiver operating characteristic (ROC) curve is established. In machine learning, TPR is sometimes referred to as recall or probability of detection. The FPR and TPR have disappeared from the left side of the ROC (see Figure 4). There are a lot of meaningful test findings on the threshold line. Start with the most recent findings, which represent the most meaningful test results. The consistency with which a measure categorizes knowledge into these two categories is quantifiable and informative. The study described here emphasizes the importance of specificity over recall (also known as responsiveness or TPR) because low precision

leaves patients with no need for therapeutic intervention. On the other hand, recall should not be neglected as a false positive result could lead to unnecessarily treating the individual. Evaluation of the models included recall and specificity, fivefold cross-validated accuracy, F1-score, and Matthew’s correlation coefficient (MCC). The accuracy is determined by the number of correctly detected observational events (both benign and malignant tumors); F1 represents the harmonic mean of precision and recall and signifies the model’s ability to predict. In other words, precision refers to the proportion of accurately detected malignancies in the anticipated set to the total number of malignancies. Each of recall, specificity, accuracy, and precision has a value between



TABLE 2: The presented CNN architecture layers.

Layer	Name	Description
1	Image input	256 × 256 × 1 images with “zero center” normalization
2	Convolution	8 (3 × 3) convolutions with stride [1 1] and padding “same”
3	Batch normalization	Normalization
4	ReLU	Rectifier
5	Pooling	2 × 2 max pooling with stride [2 2] and padding [0 0 0 0]
6	Convolution	16 (3 × 3) convolutions with stride [1 1] and padding “same”
7	Batch normalization	Normalization
8	ReLU	Rectifier
9	Pooling	2 × 2 max pooling with stride [2 2] and padding [0 0 0 0]
10	Convolution	32 (3 × 3) convolutions with stride [1 1] and padding “same”
11	Batch normalization	Normalization
12	ReLU	Rectifier
13	Fully connected	7 fully connected layers
14	SoftMax	SoftMax
15	Classification output	Cross entropy



FIGURE 6: The confusion matrix of the deep learning methods used for breast cancer diagnosis.

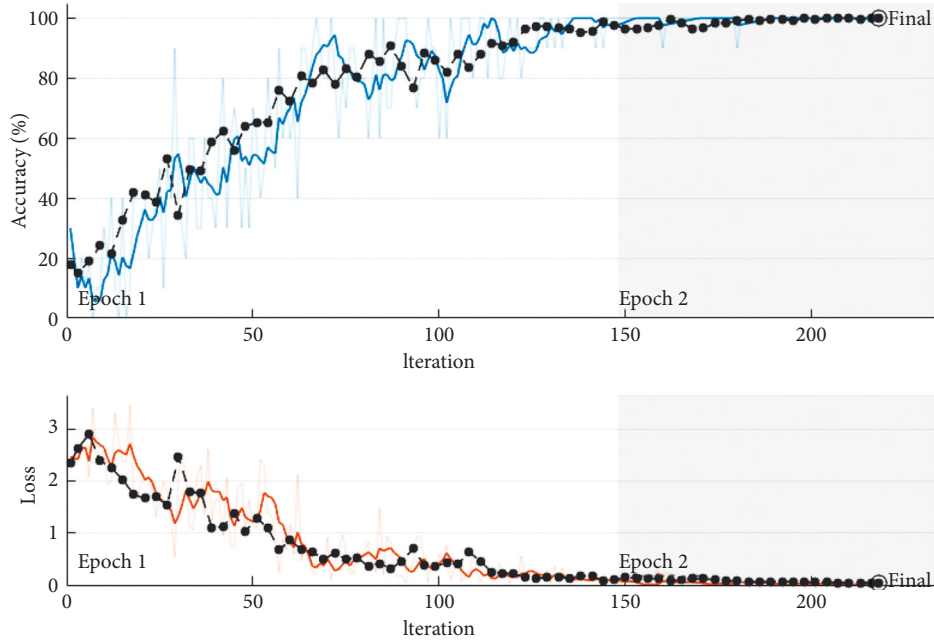


FIGURE 7: The training process of the CNN approach.

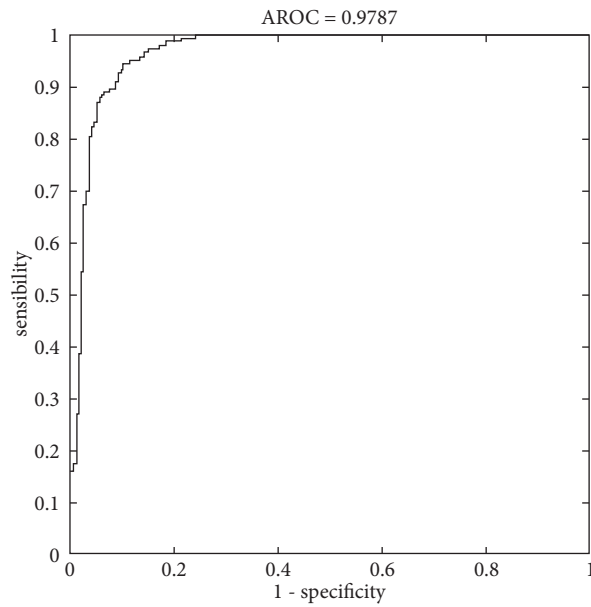


FIGURE 8: ROC region for the breast tumor segmentation in the MIAS dataset.

0 and 1. The MCC was evaluated as a metric for assessing classification accuracy, since it takes into account both true negatives as well as true positives, false positives, and false negatives, and when classes are imbalanced; it can measure classification accuracy more accurately. The most commonly used binary classification evaluation measures are accuracy

and F1-score when assessing models' performance on un-balanced datasets. It is only when a model predicts accurately in all four confusion matrix categories (true positives, false negatives, and false positives) that it is considered high in the MCC for binary classification. MCC provides more accurate estimates of classification accuracy when applied to the

TABLE 3: The comparison between the presented methods.

Methods	Sensitivity (%)	Specificity (%)	Precision (%)	AUC	Accuracy (%)
KNN	77.9	77.4	79.7	79.43	77.5
SVM	78.5	75.2	75.7	78.54	79.5
PSOWNNs	91.6	98.8	98.6	95.43	95.2
CNN	94.3	93.4	94.9	93.65	93.8

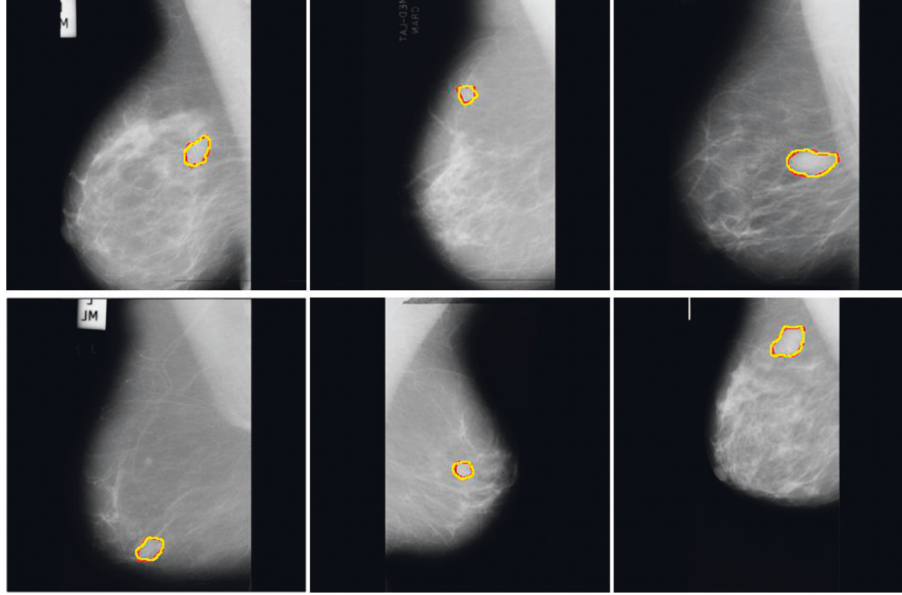


FIGURE 9: CNN for Mini-MIAS dataset was used to visually assess breast tumor segmentation.

WBCD dataset. There is a perfect prediction on an MCC score of 1 if predictions and observations do not agree. The formulas for recall, specificity, accuracy, precision, and F1-score are listed as follows.

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{Specificity} = \frac{TN}{TN + FP}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$F_1 = \frac{2TP}{2TP + FN + FP}$$

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (1)$$

A true positive, a true negative, a false positive, and a false negative, respectively, are represented in the above equations. Mixture-effect models were run in JMP to determine the effects of dimensionality reduction, outlier analysis, or combinations of both on cross-validated accuracy.

**4.7. CNN Neural Network Structure.** CNN has been a major factor in deep learning's recent success. The convolutional layers in this neural network are completely linked to the top layer. Additionally, the weights and layers are combined in this approach. The performance of this deep neural network was better than previous deep neural network designs. Furthermore, deep-feed neural networks are simpler to train. They are useful because they have a limited number of estimated parameters. Convolutional neural networks consist of three primary layers: the convolutional layer, the pooling layer, and the fully connected layer, each of which performs a unique function. There are two steps to training convolutional neural networks: feedforward and backpropagation. After feeding the input image into the neural network, the network output is calculated, which is used to adjust the network training parameters; after calculating the output, the results are used to calculate error rates for the network. Starting with the error value computed in the previous phase, backpropagation begins. The gradients of each parameter are computed using the chain rule, and each parameter varies in response to its influence on the network's error. Afterward, the feedforward phase begins. After a certain number of iterations, the training concludes. Our structure of the convolutional network is illustrated in Figure 5 and Table 2. There are twenty layers in the structure of CNN.

## 5. Results and Discussion

To evaluate, we have divided the test data into two categories, benign and malignant. The evaluation was performed on 64 samples from the benign class and 51 representatives from the malignant class from the MIAS dataset. The test data related to the benign class and the probability of belonging are considered first. The test data associated with the malignant class and its chance are considered in the second category. We now discuss the ROC, confusion matrix, and the comparison diagrams of the two classes. Figure 6 shows the ROC of the MLP with GLCM features, as well as CNN for the benign and malignant class.

The techniques are also included in each class of 905 matrices with 22 features. Figure 6 shows the classification results in a confusion matrix. Hence, in the confusion matrix in Figure 6, green cells indicate real metrics, while white cells indicate false metrics. In findings, out of 905 images of breast cancer, 830 (47.2 percent) are diagnosed correctly. Despite this, 70 (4.1 percent of all images) are misdiagnosed. So PSOWNNs techniques have a sensitivity of 91.6%. In other words, 91.6 percent of individuals with breast cancer are identified correctly. Moreover, 905 images with different disorders were correctly diagnosed in 98.8 percent of the cases. PSOWNNs have a specificity of 98.8%. Additionally, PSOWNNs have an accuracy of 98.6%, which means 830 patients are found to have breast cancer, they (98.6%) are correctly diagnosed. The accuracy of classifiers is another significant parameter. PSOWNNs have an accuracy of 95.2%. That means that 95.2% of the images are correctly diagnosed. In comparison with other machine learning techniques, PSOWNNs are more accurate than SVM, KNN, and CNN. As neural networks do, CNNs train to build output maps by removing more complex, high-level functions. Input function maps are combined with convolutional kernels. CNN exploits the fact that a feature is the same in the receptive field irrespective of its location if a function converts. The results show that CNN can acquire more useful functionality than techniques that do not take this into account. As a result of this assumption, weight sharing is used to decrease the number of components.

The used methods are trained by gradient descent. Hierarchical characteristics are optimized for the task at hand because each layer feeds into the previous one. A real-valued vector is typically needed for SVM and other methods. By contrast, CNN is often taught from beginning to end, ensuring that it reacts appropriately to the challenge it is attempting to resolve. In SVMs, KNNs, and CNNs, PSOWNNs are used as trainable attribute detectors. Since SVMs are still extensively used, different machine learning algorithms should complement each other. Consequently, this article uses machine learning to identify breast cancer based on GLCM traits. CNN's confusion matrix is depicted in Figure 5. 905 breast cancer patients were accurately diagnosed by 799 (94.9 percent) according to the matrix. However, 106 images were misdiagnosed. On the other hand, all patients with additional disorders or negative test results are identified. The CNN approaches have a sensitivity of 94.3 percent and a specificity of 93.8 percent, respectively.

They also have a 93.4 percent accuracy rate. Breast cancer is a true positive in all patients with this identification. CNN methodologies, therefore, are 95.9% accurate.

In terms of classification, it is preferable if the fall-out and sensitivity have lower and higher values, respectively. CNN has, therefore, a greater sensitivity than other methods. Meanwhile, KNN demonstrated that machine learning algorithms are less sensitive than CNN (see Figure 7). In Figure 8, the ROC curve is displayed to compare machine learning algorithms. Fall-out is represented by the horizontal axis of the ROC curve, while sensitivity is represented by the vertical axis.

Based on the performance analysis metrics results shown in Table 3, the higher values belong to the CNN technique. ROC curve value is an essential metric for classifiers. For CNN methods, it is 99.97. Consequently, the highest accuracy belongs to CNN, KNN, LDA, SVM, and NB, respectively.

## 6. Discussions

The improved SVM achieves the lowest score for all criteria on the Mini-MIAS mammography dataset. ANNs are also derived from KNNs and CNNs, but their major drawbacks are overfitting, hence the need for more training data, as well as their inability to extract features. Among the most sensitive and specific structures are machine learning methods. As a result of their less sensitivity to variance in input samples, convolutional neural networks outperform CNNs and PSOWNNs. Results show that the PSOWNNs have the best classification performance and lowest error rate. Based on the Mini-MIAS datasets, Figure 8 shows the receiver operating characteristics (ROC) area for the used structure. ROC analysis involves analyzing classification jobs at different threshold values. A statistical model's accuracy and, more broadly, the performance of a system can be evaluated based on this evaluation.

From the Mini-MIAS mammography dataset shown in Figure 9, you can see a visual representation of how the technique is used to segment different tissues and tumors.

## 7. Conclusion

A complicated illness, breast cancer is recognized as the second leading cause of cancer-associated mortality in women. A growing body of data suggests that many factors (e.g., genetics and environment) might play a role in breast cancer onset and development. Image-based diagnostic methods are among the many diagnostic platforms that can provide valuable information on breast cancer patients. Unfortunately, it is not always easy to identify breast abnormalities. Various technologies have been developed to screen for breast cancer, such as mammography, ultrasound, and thermography. Utilizing image processing and artificial intelligence (AI) techniques, the computer helps radiologists identify chest problems more effectively. This article evaluated many approaches to detecting breast cancer using AI and image processing. Using machine learning methods for identifying breast cancer, the authors present an innovative

approach. Based on the support value on a neural network, the suggested method differs from previous techniques. A normalizing technique has been implemented to benefit the image in terms of performance, efficiency, and quality. According to our experiments, PSOWNNs are relatively superior to current approaches such as CNN. There is no doubt that machine learning methods are beneficial in terms of performance, efficiency, and quality of images, which are vital to the newest medical applications. Based on the results, PSOWNNs approaches are 91.6% sensitive. That is, 91.6 percent of breast cancer patients are detected correctly. A further comparison of the process 905 image with those of other illnesses reveals that 98.6 percent of the disorders are correctly diagnosed. Therefore, PSOWNNs have a specificity of 98.8%. Furthermore, PSOWNNs have a precision of 98.6 percent, which means 830 people are seen to have breast cancer, but only 830 (95.2 percent) are diagnosed. In other words, 95.2% of images are correctly classified. PSOWNNs are more accurate than other machine learning algorithms, SVM, KNN, and CNN.

## Nomenclature

ANN:	Artificial neural network
BIRADS:	Breast Imaging Reporting and Dated System
CNN:	Convolutional neural network
DCNN:	Deep convolutional neural network
DL:	Deep learning
FN:	False negative
FP:	False positive
FPR:	False positive rate
GLCM:	Gray-level cooccurrence matrix
KNN:	k-nearest neighbor
LDA:	Linear discrimination analysis
MCC:	Matthew's correlation coefficient
ML:	Machine learning
NB:	Naïve bayes
PCA:	Principal component analysis
PSOWNN:	Particle swarm optimized wavelet neural network
ReLU:	Rectified linear unit
ROC:	Receiver operating characteristic
ROI:	Region-of-Interest
SVM:	Support vector machine
TN:	True negative
TP:	True positive
TPR:	True positive rate.

## Data Availability

Data are available and can be provided over the emails querying directly to the corresponding author (amin.valizadeh@mail.um.ac.ir).

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] M. Malvezzi, G. Carioli, P. Bertuccio et al., "European cancer mortality predictions for the year 2019 with focus on breast cancer," *Annals of Oncology*, vol. 30, no. 5, pp. 781–787, 2019.
- [2] C. Salguero-Aranda, D. Sancho-Mensat, B. Canals-Lorente, S. Sultan, A. Reginald, and L. Chapman, "STAT6 knockdown using multiple siRNA sequences inhibits proliferation and induces apoptosis of human colorectal and breast cancer cell lines," *PLoS One*, vol. 14, no. 5, Article ID e0207558, 2019.
- [3] L. K. Dunnwald, M. A. Rossing, and C. I. Li, "Hormone receptor status, tumor characteristics, and prognosis: a prospective cohort of breast cancer patients," *Breast Cancer Research*, vol. 9, no. 1, pp. R6–R10, 2007.
- [4] A. McGuire, J. A. L. Brown, and M. J. Kerin, "Metastatic breast cancer: the potential of miRNA for diagnosis and treatment monitoring," *Cancer & Metastasis Reviews*, vol. 34, no. 1, pp. 145–155, 2015.
- [5] O. Peart, "Metastatic breast cancer," *Radiologic Technology*, vol. 88, no. 5, pp. 519M–539M, 2017.
- [6] E. Zarychta and B. Ruzkowska-Ciastek, "Cooperation between angiogenesis, vasculogenesis, chemotaxis, and coagulation in breast cancer metastases development: pathophysiological point of view," *Biomedicines*, vol. 10, no. 2, p. 300, 2022.
- [7] A. Farrokh, G. Goldmann, U. Meyer-Johann et al., "Clinical differences between invasive lobular breast cancer and invasive carcinoma of no special type in the german mammography-screening-program," *Women & Health*, vol. 62, pp. 1–13, 2022.
- [8] E. R. Fisher, R. M. Gregorio, C. Redmond, and B. Fisher, "Tubulolobular invasive breast cancer: a variant of lobular invasive cancer," *Human Pathology*, vol. 8, no. 6, pp. 679–683, 1977.
- [9] J. Kotsopoulos, W. Y. Chen, M. A. Gates, S. S. Tworoger, S. E. Hankinson, and B. A. Rosner, "Risk factors for ductal and lobular breast cancer: results from the nurses' health study," *Breast Cancer Research*, vol. 12, no. 6, pp. R106–R111, 2010.
- [10] Y. Guan, R. Haardörfer, C. M. McBride, J. Lipscomb, and C. Escoffery, "Factors associated with mammography screening choices by women aged 40-49 at average risk," *Journal of Women's Health*, 2022.
- [11] W. Ding and J. X. Zhang, "Deep transfer learning-based intelligent diagnosis of malignant tumors on mammography," in *Proceedings of the 2021 3rd International Conference on Industrial Artificial Intelligence (IAI)*, pp. 1–6, IEEE, Shenyang China, 2021 November.
- [12] Z. Rezaei, "A review on image-based approaches for breast cancer detection, segmentation, and classification," *Expert Systems with Applications*, vol. 182, Article ID 115204, 2021.
- [13] C. Kaur and U. Garg, "Artificial intelligence techniques for cancer detection in medical image processing: a review," *Materials Today Proceedings*, 2021.
- [14] A. Hamidinekoo, E. Denton, A. Rampun, K. Honnor, and R. Zwigelaar, "Deep learning in mammography and breast histology, an overview and future trends," *Medical Image Analysis*, vol. 47, pp. 45–67, 2018.
- [15] S. Ekici and H. Jawzal, "Breast cancer diagnosis using thermography and convolutional neural networks," *Medical Hypotheses*, vol. 137, Article ID 109542, 2020.
- [16] M. W. Retsky, R. Demicheli, W. J. M. Hrushesky, M. Baum, and I. D. Gukas, "Dormancy and surgery-driven escape from dormancy help explain some clinical features of breast cancer," *Apmis*, vol. 116, no. 7-8, pp. 730–741, 2008.

- [17] B. Zheng, S. W. Yoon, and S. S. Lam, "Breast cancer diagnosis based on feature extraction using a hybrid of K-means and support vector machine algorithms," *Expert Systems with Applications*, vol. 41, no. 4, pp. 1476–1482, 2014.
- [18] F. Al-Turjman and T. Stephan, "An automated breast cancer diagnosis using feature selection and parameter optimization in ANN," *Computers & Electrical Engineering*, vol. 90, Article ID 106958, 2021.
- [19] A. Guha, M. G. Fradley, S. F. Dent et al., "Incidence, risk factors, and mortality of atrial fibrillation in breast cancer: a SEER-Medicare analysis," *European Heart Journal*, vol. 43, no. 4, pp. 300–312, 2022.
- [20] J. A. Cruz and D. S. Wishart, "Applications of machine learning in cancer prediction and prognosis," *Cancer Informatics*, vol. 2, Article ID 117693510600200, 2006.
- [21] F. Ahmad, N. A. Mat Isa, Z. Hussain, M. K. Osman, and S. N. Sulaiman, "A GA-based feature selection and parameter optimization of an ANN in diagnosing breast cancer," *Pattern Analysis & Applications*, vol. 18, no. 4, pp. 861–870, 2015.
- [22] W. Dong, W. P. Bensken, U. Kim, J. Rose, N. A. Berger, and S. M. Koroukian, "Phenotype discovery and geographic disparities of late-stage breast cancer diagnosis across U.S. Counties: a machine learning approach," *Cancer Epidemiology Biomarkers & Prevention*, vol. 31, no. 1, pp. 66–76, 2022.
- [23] P. Stephan, T. Stephan, R. Kannan, and A. Abraham, "A hybrid artificial bee colony with whale optimization algorithm for improved breast cancer diagnosis," *Neural Computing & Applications*, vol. 33, no. 20, pp. 13667–13691, 2021.
- [24] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," *Electronic Markets*, vol. 31, no. 3, pp. 685–695, 2021.
- [25] S. Pouyanfar, S. Sadiq, Y. Yan et al., "A survey on deep learning," *ACM Computing Surveys*, vol. 51, no. 5, pp. 1–36, 2019.
- [26] S. Dargan, M. Kumar, M. R. Ayyagari, and G. Kumar, "A survey of deep learning and its applications: a new paradigm to machine learning," *Archives of Computational Methods in Engineering*, vol. 27, no. 4, pp. 1071–1092, 2020.
- [27] S. M. Lee, J. B. Seo, J. Yun et al., "Deep learning applications in chest radiography and computed tomography," *Journal of Thoracic Imaging*, vol. 34, no. 2, pp. 75–85, 2019.
- [28] S. Beura, B. Majhi, and R. Dash, "Mammogram classification using two dimensional discrete wavelet transform and gray-level co-occurrence matrix for detection of breast cancer," *Neurocomputing*, vol. 154, pp. 1–14, 2015.
- [29] J. P. Liu and C. L. Li, "The short-term power load forecasting based on sperm whale algorithm and wavelet least square support vector machine with DWT-IR for feature selection," *Sustainability*, vol. 9, no. 7, p. 1188, 2017.
- [30] U. R. Acharya, U. Raghavendra, H. Fujita et al., "Automated characterization of fatty liver disease and cirrhosis using curvelet transform and entropy features extracted from ultrasound images," *Computers in Biology and Medicine*, vol. 79, pp. 250–258, 2016.
- [31] D. Muduli, R. Dash, and B. Majhi, "Fast discrete curvelet transform and modified PSO based improved evolutionary extreme learning machine for breast cancer detection," *Bio-medical Signal Processing and Control*, vol. 70, Article ID 102919, 2021.
- [32] S. Khan, M. Hussain, H. Aboalsamh, H. Mathkour, G. Bebis, and M. Zakariah, "Optimized Gabor features for mass classification in mammography," *Applied Soft Computing*, vol. 44, pp. 267–280, 2016.
- [33] Z. Xiao and Z. Yan, "Radar emitter identification based on auto-correlation function and bispectrum via convolutional neural network," *IEICE - Transactions on Communications*, vol. E104.B, no. 12, pp. 1506–1513, 2021.
- [34] R. Liu, X. Wang, H. Lu et al., "SCCGAN: style and characters inpainting based on CGAN," *Mobile Networks and Applications*, vol. 26, no. 1, pp. 3–12, 2021.
- [35] W. Zhou, L. Yu, Y. Zhou, W. Qiu, M.-W. Wu, and T. Luo, "Local and global feature learning for blind quality evaluation of screen content and natural scene images," *IEEE Transactions on Image Processing*, vol. 27, no. 5, pp. 2086–2095, 2018.
- [36] B. Li, Y. Feng, Z. Xiong, W. Yang, and G. Liu, "Research on AI security enhanced encryption algorithm of autonomous IoT systems," *Information Sciences*, vol. 575, pp. 379–398, 2021.
- [37] L. Yang, Z. Xiong, G. Liu, Y. Hu, X. Zhang, and M. Qiu, "An analytical model of page dissemination for efficient big data transmission of C-ITS," *IEEE Transactions on Intelligent Transportation Systems*, 2021.
- [38] E. Eslami and H.-B. Yun, "Attention-based multi-scale convolutional neural network (a+mcnn) for multi-class classification in road images," *Sensors*, vol. 21, no. 15, p. 5137, 2021.
- [39] L. Liao, L. Du, and Y. Guo, "Semi-supervised SAR target detection based on an improved faster R-CNN," *Remote Sensing*, vol. 14, no. 1, p. 143, 2021.
- [40] H. Liu, J. Liu, S. Hou, T. Tao, and J. Han, "Perception consistency ultrasound image super-resolution via self-supervised CycleGAN," *Neural Computing and Applications*, vol. 10, pp. 1–11, 2021.
- [41] A. Sharifi, M. Ahmadi, M. A. Mehni, S. Jafarzadeh Ghouschi, and Y. Pourasad, "Experimental and numerical diagnosis of fatigue foot using convolutional neural network," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 24, no. 16, pp. 1828–1840, 2021.
- [42] C. Zuo, J. Qian, S. Feng et al., "Deep learning in optical metrology: a review," *Light: Science & Applications*, vol. 11, no. 1, pp. 1–54, 2022.
- [43] S. He, F. Guo, Q. Zou, and H. Ding, "MRMD2.0: a python tool for machine learning with feature ranking and reduction," *Current Bioinformatics*, vol. 15, no. 10, pp. 1213–1221, 2021.
- [44] M. Ahmadi, F. Dashti Ahangar, N. Astaraki, M. Abbasi, and B. Babaei, "FWNNet: presentation of a new classifier of brain tumor diagnosis based on fuzzy logic and the wavelet-based neural network using machine-learning methods," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 8542637, 13 pages, 2021.
- [45] Q. Zou, P. Xing, L. Wei, and B. Liu, "Gene2vec: gene subsequence embedding for prediction of mammalian N6-methyladenosine sites from mRNA," *Rna*, vol. 25, no. 2, pp. 205–218, 2019.
- [46] K. Jin, Y. Yan, M. Chen et al., "Multimodal deep learning with feature level fusion for identification of choroidal neovascularization activity in age-related macular degeneration," *Acta Ophthalmologica*, vol. 100, no. 2, pp. e512–e520, 2022.
- [47] L. Yang, X. Wu, M. Luo et al., "Na<sup>+</sup>/Ca<sup>2+</sup> induced the migration of soy hull polysaccharides in the mucus layer in vitro," *International Journal of Biological Macromolecules*, vol. 199, 2022.
- [48] H. C. Chuang, D. R. Chen, and Y. L. Huang, "Using tumor morphology to classify benign and malignant solid breast masses: speckle reduction imaging (SRI) versus non-SRI ultrasound imaging. IEICE technical report," *IEICE Tech. Rep.* vol. 108, no. 385, pp. 33–36, 2009.
- [49] S. Zhou, J. Shi, J. Zhu, Y. Cai, and R. Wang, "Shearlet-based texture feature extraction for classification of breast tumor in

- ultrasound image,” *Biomedical Signal Processing and Control*, vol. 8, no. 6, pp. 688–696, 2013.
- [50] Y. Liu, R. Zou, and H. Guo, “Risk explicit interval linear programming model for uncertainty-based nutrient-reduction optimization for the lake Qionghai watershed,” *Journal of Water Resources Planning and Management*, vol. 137, no. 1, pp. 83–91, 2011.
- [51] J. Zuluaga-Gomez, Z. Al Masry, K. Benagoune, S. Meraghni, and N. Zerhouni, “A CNN-based methodology for breast cancer diagnosis using thermal images,” *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, vol. 9, no. 2, pp. 131–145, 2021.
- [52] E. D. Carvalho, O. C. Antonio Filho, R. R. Silva et al., “Breast cancer diagnosis from histopathological images using textural features and CBIR,” *Artificial Intelligence in Medicine*, vol. 105, 2020.
- [53] T. Mahmood, J. Li, Y. Pei, F. Akhtar, Y. Jia, and Z. H. Khand, “Breast mass detection and classification using deep convolutional neural networks for radiologist diagnosis assistance,” in *Proceedings of the 2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC)*, pp. 1918–1923, IEEE, Madrid, Spain, 2021, July.
- [54] Z. Zhang, P. Ma, R. Ahmed et al., “Advanced point-of-care testing technologies for human acute respiratory virus detection,” *Advanced Materials*, vol. 34, no. 1, Article ID 2103646, 2022.
- [55] J. Jiang, Y. Guo, Z. Huang, Y. Zhang, D. Wu, and Y. Liu, “Adjacent surface trajectory planning of robot-assisted tooth preparation based on augmented reality,” *Engineering Science and Technology: An International Journal*, vol. 27, Article ID 101001, 2022.
- [56] C. Qin, D. Xiao, J. Tao et al., “Concentrated velocity synchronous linear chirplet transform with application to robotic drilling chatter monitoring,” *Measurement*, vol. 194, Article ID 111090, 2022.
- [57] L. Mobasheri, M. H. Nasirpour, E. Masoumi, A. F. Azarnaminy, M. Jafari, and S.-A. Esmaeili, “SARS-CoV-2 triggering autoimmune diseases,” *Cytokine*, vol. 154, Article ID 155873, 2022.
- [58] A. Ala, FE Alsaadi, M. Ahmadi, and S. Mirjalili, “Optimization of an appointment scheduling problem for healthcare systems based on the quality of fairness service using whale optimization algorithm and NSGA-II,” *Scientific Reports*, vol. 11, no. 1, pp. 19816–19819, 2021.
- [59] Z. Liu, L. Fang, D. Jiang, and R. Qu, “A machine-learning based fault diagnosis method with adaptive secondary sampling for multiphase drive systems,” *IEEE Transactions on Power Electronics*, 2022.
- [60] W. Zheng, X. Liu, and L. Yin, “Research on image classification method based on improved multi-scale relational network,” *PeerJ Computer Science*, vol. 7, p. e613, 2021.
- [61] Y. Liu, J. Tian, R. Hu et al., “Improved feature point pair purification algorithm based on SIFT during endoscope image stitching,” *Frontiers in Neurorobotics*, vol. 16, Article ID 840594, 2022.
- [62] P. Kaur, G. Singh, and P. Kaur, “Intellectual detection and validation of automated mammogram breast cancer images by multi-class SVM using deep learning classification,” *Informatics in Medicine Unlocked*, vol. 16, Article ID 100239, 2019.
- [63] C. El Chamieh, P. Vielh, and S. Chevret, “Statistical methods for evaluating the fine needle aspiration cytology procedure in breast cancer diagnosis,” *BMC Medical Research Methodology*, vol. 22, no. 1, pp. 40–11, 2022.
- [64] T. Rajkumar, S. Amritha, V. Sridevi et al., “Identification and validation of plasma biomarkers for diagnosis of breast cancer in South Asian women,” *Scientific Reports*, vol. 12, no. 1, pp. 100–113, 2022.
- [65] S. R. Sannasi Chakravarthy, N. Bharanidharan, and H. Rajaguru, “Multi-deep CNN based experimentations for early diagnosis of breast cancer,” *IETE Journal of Research*, pp. 1–16, 2022.
- [66] Z. Wang, H. Sun, J. Li et al., “Preoperative prediction of axillary lymph node metastasis in breast cancer using CNN based on multiparametric MRI,” *Journal of Magnetic Resonance Imaging*, 2022.
- [67] J. G. Melekoodappattu, A. S. Dhas, B. K. Kandathil, and K. S. Adarsh, “Breast cancer detection in mammogram: combining modified CNN and texture feature based approach,” *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–10, 2022.
- [68] C. B. Gonçalves, J. R. Souza, and H. Fernandes, “CNN architecture optimization using bio-inspired algorithms for breast cancer detection in infrared images,” *Computers in Biology and Medicine*, vol. 142, Article ID 105205, 2022.
- [69] J. Li, G. Yuan, and H. Fan, “Multifocus image fusion using wavelet-domain-based deep CNN,” *Computational intelligence and neuroscience*, vol. 2019, Article ID 4179397, 2019.
- [70] Z. Zhang, L. Wang, W. Zheng, L. Yin, R. Hu, and B. Yang, “Endoscope image mosaic based on pyramid ORB,” *Biomedical Signal Processing and Control*, vol. 71, no. 2022, Article ID 103261.
- [71] S. Liu, B. Yang, Y. Wang, J. Tian, L. Yin, and W. Zheng, “2D/3D multimode medical image registration based on normalized cross-correlation,” *Applied Sciences*, vol. 12, no. 6, p. 2828, 2022.
- [72] Q. Xu, Y. Zeng, W. Tang et al., “Multi-task joint learning model for segmenting and classifying tongue images using a deep neural network,” *IEEE journal of biomedical and health informatics*, vol. 24, no. 9, pp. 2481–2489, 2020.
- [73] M. Ahmadi, A. Sharifi, M. Jafarian Fard, and N. Soleimani, “Detection of brain lesion location in MRI images using convolutional neural network and robust PCA,” *International Journal of Neuroscience*, pp. 1–12, 2021.
- [74] E. Sadeghipour, N. Sahragard, M.-R. Sayebani, and R. Mahdizadeh, “Breast cancer detection based on a hybrid approach of firefly algorithm and intelligent systems,” *Indian Journal of Fundamental and Applied Life Sciences*, vol. 5, p. S1, 2015.
- [75] M. Zhang, Y. Chen, and W. Susilo, “PPO-CPQ: a privacy-preserving optimization of clinical pathway query for e-healthcare systems,” *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 10660–10672, 2020.
- [76] E. Sadeghipour, A. Hatam, and F. Hosseinzadeh, “Designing an intelligent system for diagnosing diabetes with the help of the Xcsla system,” *Journal of Mathematics and Computer Science*, vol. 14, no. 1, pp. 24–32, 2015.
- [77] M. Ahmadi, A. Sharifi, S. Hassantabar, and S. Enayati, “QAIS-DSNN: tumor area segmentation of MRI image with optimized quantum matched-filter technique and deep spiking neural network,” *BioMed Research International*, vol. 202116 pages, 2021.
- [78] J. Chen, D. Lan, and Y. Guo, “Label constrained convolutional factor analysis for classification with limited training samples,” *Information Sciences*, vol. 544, pp. 372–394, 2021.
- [79] M. Rezaei, F. Farahanipad, A. Dillhoff, R. Elmasri, and V. Athitsos, “Weakly-supervised hand part segmentation from depth images,” in *Proceedings of the 14th Pervasive*

- Technologies Related to Assistive Environments Conference*, pp. 218–225, Corfu, Greece, June 2021.
- [80] E. Sadeghipour, A. Hatam, and F. Hosseinzadeh, “An expert clinical system for diagnosing obstructive sleep apnea with help from the XCSR classifier,” *Journal of Mathematics and Computer Science*, vol. 14, no. 1, pp. 33–41, 2015.
- [81] S. Dorosti, S. Jafarzadeh Ghoushchi, E. Sobhrakhshankhah, M. Ahmadi, and A. Sharifi, “Application of gene expression programming and sensitivity analyses in analyzing effective parameters in gastric cancer tumor size and location,” *Soft Computing*, vol. 24, no. 13, pp. 9943–9964, 2020.
- [82] M. Q. H. Abadi, S. Rahmati, A. Sharifi, M. Ahmadi, and M. Ahmadi, “HSSAGA: designation and scheduling of nurses for taking care of COVID-19 patients using novel method of hybrid salp swarm algorithm and genetic algorithm,” *Applied Soft Computing*, vol. 108, Article ID 107449, 2021.
- [83] B. Zhou and B. Arandian, “An Improved CNN Architecture to Diagnose Skin Cancer in Dermoscopic Images Based on Wildebeest Herd Optimization Algorithm,” *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 7567870, 2021.
- [84] A Davoudi, M Ahmadi, A Sharifi et al., “Studying the effect of taking statins before infection in the severity reduction of COVID-19 with machine learning,” *BioMed Research International*, vol. 2021, Article ID 9995073, 12 pages, 2021.
- [85] S. Hassantabar, M. Ahmadi, and A. Sharifi, “Diagnosis and detection of infected tissue of COVID-19 patients based on lung X-ray image using convolutional neural network approaches,” *Chaos, Solitons & Fractals*, vol. 140, Article ID 110170, 2020.
- [86] Z. Yu, Z. Wang, X. Yu, and Z. Zhang, “RNA-seq-based breast cancer subtypes classification using machine learning approaches,” *Computational intelligence and neuroscience*, vol. 2020, Article ID 4737969, 2020.
- [87] M. Thilagaraj, N. Arunkumar, and P. Govindan, “Classification of breast cancer images by implementing improved dcnn with artificial fish school model,” *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 6785707, 2022.



## Research Article

# FDCNet: Presentation of the Fuzzy CNN and Fractal Feature Extraction for Detection and Classification of Tumors

Sepideh Molaei,<sup>1</sup> Niloofar Ghorbani ,<sup>2</sup> Fatemeh Dashtiahangar,<sup>3</sup>  
Mohammad Peivandi ,<sup>4</sup> Yaghoub Pourasad ,<sup>5</sup> and Mona Esmaeili<sup>6</sup>

<sup>1</sup>Faculty of Electrical and Computer Engineering, Tarbiat Modares University, Tehran, Iran

<sup>2</sup>High Point University, Department of Mathematical Sciences, High Point, NC, USA

<sup>3</sup>Department of Electrical Engineering, Golestan University, Gorgan, Iran

<sup>4</sup>Fachbereich Wirtschafts- und Rechtswissenschaften, HTW Berlin - University of Applied Sciences, Berlin, Germany

<sup>5</sup>Department of Electrical Engineering, Urmia University of Technology, Urmia, Iran

<sup>6</sup>Department of Electrical and Computer Engineering, University of NM, Albuquerque, NM 8731, USA

Correspondence should be addressed to Yaghoub Pourasad; y.pourasad@uut.ac.ir

Received 17 February 2022; Accepted 8 April 2022; Published 6 May 2022

Academic Editor: Mohamed Abdelaziz

Copyright © 2022 Sepideh Molaei et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The detection of brain tumors using magnetic resonance imaging is currently one of the biggest challenges in artificial intelligence and medical engineering. It is important to identify these brain tumors as early as possible, as they can grow to death. Brain tumors can be classified as benign or malignant. Creating an intelligent medical diagnosis system for the diagnosis of brain tumors from MRI imaging is an integral part of medical engineering as it helps doctors detect brain tumors early and oversee treatment throughout recovery. In this study, a comprehensive approach to diagnosing benign and malignant brain tumors is proposed. The proposed method consists of four parts: image enhancement to reduce noise and unify image size, contrast, and brightness, image segmentation based on morphological operators, feature extraction operations including size reduction and selection of features based on the fractal model, and eventually, feature improvement according to segmentation and selection of optimal class with a fuzzy deep convolutional neural network. The BraTS data set is used as magnetic resonance imaging data in experimental results. A series of evaluation criteria is also compared with previous methods, where the accuracy of the proposed method is 98.68%, which has significant results.

## 1. Introduction

The soft and spongy mass of tissue protected by the cranial bone is called the brain, which has three thin layers of tissue (meninges). A watery fluid circulates in the spaces inside the brain (Cerebral Spinal Fluid). The brain controls both voluntary (such as walking or speaking) and automatic activities (such as breathing). The brain is responsible for memory, emotions, and personality in addition to the five senses (sight, hearing, touch, taste, and smell). An intricate network of nerves connects the brain and body. A benign brain tumor does not contain cancer cells, but a malignant brain tumor does. It is vital in the health system to identify

and diagnose brain tumors. Early detection of cancer can help a person live longer. The brain has a variety of tissues that result from the accumulation and overgrowth of cells to cause brain tumors. Some of the symptoms of brain tumors are headaches, seizures, mood swings, and movement disorders. There are three treatments for brain tumors: surgery, radiation therapy, and chemotherapy. Each symptom has side effects that have been studied in [1].

MRI images are widely used due to their excellent spatial resolution and tissue contrast. The extent to which a tumor threatens depends on different factors such as the type, the location and size of the tumor, and how it spreads and develops. MRI elements must be divided into four categories

to diagnose: white matter (WM), gray matter (GM), Cerebral Spinal Fluid (CSF), and abnormal brain tissue (tumor). The purpose of brain tumor segmentation is to isolate various tumor tissues such as active cells, dead tissue center, swelling in normal white matter (WM), gray matter (GM), and Cerebral Spinal Fluid (CSF). Due to the high medical diagnosing capabilities of computer science, such as image processing and machine vision, these have been used extensively in various fields of science. This study provides a comprehensive approach to the diagnosis and classification of benign and malignant brain tumors. Studies and innovations of the proposed method are included in the following:

- (1) A fuzzy deep convolutional neural network method called FDCNet is proposed for diagnosing and classifying brain tumors using MRI images. The FDCNet approach integrates fuzzy theory with the remaining network, generating fuzzy rules using an adaptive learning algorithm. It examines fuzzy and deep indexes to construct the features of brain tumors and also solve the problem of uncertainty.
- (2) A deep fuzzy convolutional network model is designed to detect and classify brain tumors from MRI images using the representation of tensor data to examine the temporal and locative features of brain tumors. FDCNet structure is optimized by exploring the number of layers in the regression model and functions. The FDCNet method has high identification, detection, and classification capabilities.
- (3) The proposed approach in the BraTS data set is trained and tested to compare and observe the assurance that the proposed method can measure in terms of evaluation criteria. The main advantage of MRI imaging detection and classification systems on brain tumors is the improvement of evaluation criteria in the same conditions of the data set with the presented methods that can be used to determine the efficiency of the methods.

In this paper, the following sections are presented: (1) "Introduction" section describes the problem, details the contribution, and shows the novelty of the approach. (2) "Literature Review" section provides information on recent research investigating the problem and the methodology. (3) "Methods and Materials" section provides an overview of the approach characteristics and the strategies proposed. (4) "Results and Discussion" section aims to analyze the prediction outcome using the provided technique. (5) "Conclusion" section concludes the presentation by providing an overview of the overall results and future directions.

## 2. Literature Review

Various studies have been conducted on brain tumors diagnosis. The fragmentation has been discussed as a deep learning technique to detect brain tumors from MRI images using the convolutional neural network in [2]. The BraTS

data set has been considered as the input data. This study indicates an improvement in fragmentation to classify and detect brain tumors from data sets with relatively high accuracy. Another study [3] discussed fragmentation operations to identify and classify brain tumors from MRI images using a generative adversarial network (GAN). This method combines the production model with the discriminatory model and uses the generative adversarial network instead of a conditional random field as a high-level smoothing method. The proposed method was trained and tested in the BraTS data set. It can be demonstrated that the proposed method gains a competitive result and the usage of a generative adversarial network improves the network's performance.

Furthermore, this approach only takes about 10.8 seconds to perform high-precision segmentation and brain tumors diagnosis compared to other preconvolutional neural network-based methods. The average accuracy of this method has been 94.5%. Similarly, a generative adversarial network (GAN) in [4] has also been used for the segmentation, detection, and classification operation of brain tumors from MRI images. The optimization operation of the generative adversarial neural network has been performed using a method called progressively growing GAN named PGGAN. The main features of this method were as follows: considering the size of the images as  $256 \times 256$  pixels in an integrated form, also illuminance intensity, and the position of the masses and edges, where the average accuracy of this method was 95%. Using a convolutional neural network with a combinational method called neutrosophic expert maximum fuzzy sure entropy has been studied to detect and classify brain tumors from MRI images in [5]. Liu et al. (2021) suggested a new method to produce super-resolution (SR) ultrasound images that maintains perceptual consistency. The PSNR/IFC/SSIM, inference efficiency, and visual impacts evaluations using the benchmark data sets CCA-US and CCA-US demonstrate that our proposed methodology is both effective and superior to the existing state-of-the-art approaches [6]. According to Zhang et al. (2020), a new method is proposed for optimizing clinical pathway queries in e-healthcare systems while preserving privacy [7]. Tang et al. (2018) described a technique for identifying the origin of tumors using tissue-specific miRNA and DNA methylation indicators [8]. In the general approach called NS-CNN-EMFSE, the final classification uses a support vector machine and  $K$ -Nearest Neighbor method.  $K$ -fold has also been used for cross-validation and accuracy for evaluation. BraTS data sets are considered input data, and the results show that the classification accuracy is 96.52%. Also, deep neural network convolution is used by amplifying extensive data in the diagnosis of brain tumors from MRI images and BraTS data sets in [9]. Extensive data amplification has been considered to improve diagnostic and classification operations. The accuracy of the diagnosis and classification results was 95.5%. Liu et al. (2021) proposed a method for style and character inpainting. Based on the CGAN network repair style, the paper proposes font content that is similar to the correct content [10]. Chen et al. (2021) suggested using LSTM and clustering algorithms to predict human N6-

methyladenosine (m6A) sites [11]. Ahmadi et al. (2022) presented a novel model for energy forecasting that incorporates a deep convolutional neural network with fuzzy wavelets and a PSO method [12]. An expert system in [13] is considered for detecting brain tumors from MRI images. The research approach is to use the fuzzy *C*-means method or FCM in fragmentation operations and data training in the convolutional neural network. The membership functions (MFs) in fragmentation have been an effective and well-performing feature identification method. Also, another method named Extreme Learning Machine (ELM) has been used in the final classification section. The approach which uses BraTS data images as super-pixel is called SR-FCM-CNN-ELM. Its accuracy rate is 98.33%. In [14], Discrete Wavelet Transform (DWT) has been used during the training and testing of brain MRI data to classify and segment tumors in repetitive rings of the convolution neural network. Also, an innovative noise reduction method called Partial Differential Diffusion Filter (PDDF) has been proposed in this study, reducing salt and pepper noise to a large extent in the BraTS data set. The accuracy of this method to detect and classify brain tumors from MRI images in the BraTS data set is 98%. Xu et al. (2021) suggested a technique for segmenting and detecting tongue images using a deep convolutional neural network called the Multitask Joint Learning Model [15]. In [16], brain tumor detection in MRI images and also image segmentation operations using the Genetic Algorithm (GA) are presented, and the Discrete Wavelet Transform (DWT) has been used for image improvement. In another study in [17], Ant Colony Optimization (ACO) and the Genetic Algorithm (GA) have improved the implementation of MRI image fragmentation. But the general basis of brain tumor detection methods is derived from machine image processing and vision systems. Magnetic Resonance Spectroscopy (MRS) and Diffusion-Weighted Imaging (DWI) in the supratentorial area of brain tumors are among the studies conducted in this field of image processing and computer vision methods [18]. Ahmadi et al. (2021) described a unique hybrid strategy for user selection in Federated Learning using Deep-Q-Reinforcement Learning and Spectral Clustering. On the three data sets, MNIST, Fashion MNIST, and CIFAR-10, the achieved reductions in communication rounds are 51%, 25%, and 44%, respectively [19]. Eslami et al. [20] have investigated an attention-based multi-scale convolutional neural network (A+CNN) for multiclass classification in road images. Ala et al. [21] examined a variety of hypotheses to meet the analysis and different factors related to hospital patients as well as calculating WOA and NSGA. Abadi et al. [22] used a hybrid salp swarm algorithm and genetic algorithm for identifying and scheduling COVID-19 patients for care. It was shown by Mahmoudi et al. [23] that the adjusted SEQIER model is a good fit to the real COVID-induced daily death data, as it could capture the nonlinearities of the data well. Sadeghipour et al. [24] used the firefly algorithm and intelligent systems to detect breast cancer. According to Ahmadi et al. [25], they used convolutional neural networks (CNNs) to segment tumors in seven types of brain diseases: Glioma, Meningioma, Alzheimer's,

Alzheimer's plus, Pick, Sarcoma, and Huntington disease. Yazdani et al. [26] proposed in this paper an improved version of extended classification system "trained" using examples from existing data. This would help to identify and avoid attempts to interfere with computer systems during application phases. Rezaei et al. [27] proposed a method for segmenting hand parts using depth maps without any additional effort involved in obtaining segmentation labels. Mobasheri et al. [28] reviewed the immunological findings in COVID-19 and the current reports on autoimmune diseases associated with this disease. In their study, Hasantabar et al. [29] presented two algorithms, including a deep neural network (DNN) and a convolutional neural network (CNN) directly using images of the lungs. Dorosti et al. [30] presented a general model to identify the correlation of different parameters in a GC tumor place and tumor size. The medical documents of GC patients consist of the dataset of this study. Yan et al. (2020) developed a method of determining whether waxy or regular maize starches are more digestible after thermal treatment using plasma-activated water [31]. Shi et al. (2021) investigated the effects of ultrasonic intensity on wheat starch and monoglyceride combinations as well as its influence on the quality of Chinese steamed bread [32]. Yang et al. (2019) examined methods for improving the quality of HEVC-compressed videos. Finally, they created a prototype that demonstrates our TQEO concept in action [33]. The study deals with the tumors in the cerebellum region approach. Diffusion-Weighted Imaging is an imaging technique that uses the Brownian motions of water molecules to generate contrast in MRI images. Another method using edge detection is presented in [34], which has modeled the upgraded method of edge detection for segmenting the brain tumor in MRI images called the Sobel edge detection method. The steps of the segmentation algorithm have four stages: (a) finding the image gradient with the Sobel edge detector operator, (b) calculating the dependent threshold image repetition in the cycle, (c) applying the closed contour algorithm, and (d) fragmentation of objects in the image based on the pixel intensity between the closed contours. It is remarkable to use combinational simpler methods in machine vision and image processing. MRI images of brain tumors were performed by watershed segmentation and the Hierarchical Clustering Algorithm [35]. The hierarchical clustering methods used in this study include the *K*-Nearest Algorithm, the *K*-Farthest Algorithm, and the Minimal Spanning Tree method. Also, in [36], anisotropic diffusion based on fragmentation and the pattern based on group classification of MRI images to detect a brain tumor based on a support vector machine and fragmentation with FCM were performed. According to Xu et al. (2019), omnidirectional videos were assessed for visual quality. Researchers in this study demonstrate that both subjective and objective methods of video quality assessment for omnidirectional videos increase state of the art [37]. In their study, Sheng et al. (2021) examined near-online monitoring in blockchain-based edge devices with constraints on cooccurrence [38]. Neural networks have been utilized in various fields for detecting brain tumors. In [39], an unsupervised learning

method by a clustering technique for identifying a brain tumor in MRI images based on fragmentation has been proposed. Using the self-organizing mapping (SOM) neural network (NN) method with fuzzy  $K$ -means (FKM) algorithm has been the main proposed method of this research. In a similar study in [40], MRI imaging was performed to detect a brain tumor based on a neural network of self-organizing mapping and entropy-gradient clustering. The study by Chen et al. (2021) examined cellular autophagy and noncoding RNA's role in colorectal cancer [41]. He et al. (2020) explored the use of multibranch deep residual learning for clustering and beamforming in user-centric networks [42]. In a different approach in [43], MRI images' brain tumor detection and fragmentation operations have been performed using two techniques. The first one uses the Gabor filter method and Wavelet Transform. In contrast to that, statistical features methods were used, such as Linear Support Vector Machine (LSVM), Radial Basis Function (RBF),  $K$ -Nearest Neighbor (KNN), Euclidean  $K$ -means and blocked  $K$ -means, and Sparse Representation. Combining neural networks with other methods is also common in tumor diagnosis [44], which provides the performance analysis of a classification method for identifying and detecting brain tumors from MRI images based on the neural-fuzzy network [45]. Niu et al. (2021) used ensembles of convolutional neural networks to detect sgRNA on-target activity in four crops. These findings have implications for agricultural gene editing and academic research [46]. Sun et al. (2021) evaluated a babysitting package for the analysis of retrospective and freshly produced RNA-seq data using both alignment-based and alignment-free quantification methods [47]. Zhang et al. (2020) examined the privacy-preserving clinical pathway query optimization for healthcare networks [7]. Olowookere [48] created two- and three-class models for breast cancer detection and classification using a deep convolutional neural network and fuzzy support vector machines. According to this study's addition to knowledge, the hybridization of deep convolutional neural networks with fuzzy support vector machines improved the identification of malignant and noncancerous breast cancers in both binary and three-class classification situations. Ragab et al. [49] created a novel ensemble deep-learning-enabled clinical decision support system that uses ultrasound pictures to diagnose and classify breast cancer. The researchers devised an optimum multilevel thresholding-based picture segmentation approach to detect the tumor-affected areas. The researchers also created a feature extraction ensemble comprising three deep learning models and an effective machine learning classifier for breast cancer diagnosis. The research provides a method for radiologists and healthcare practitioners to aid in the categorization of breast cancer [49]. Liu et al. (2021) investigated perception constancy ultrasound image super-resolution using self-supervised CycleGAN. Using the benchmark data sets CCA-US and CCA-US, we evaluated PSNR/IFC/SSIM, inference efficiency, and visual impact of our proposed approach for comparison with existing state-of-the-art approaches [6]. Tang et al. (2018) examined the possibility of detecting tumor origins using tissue-specific miRNA and DNA

methylation indicators. They created a user-friendly website that enables users to determine the origin of tumors by uploading miRNA or DNAm profiles relevant to their research [8]. Sharifi et al. (2021) used convolutional neural networks to investigate the experimental and numerical diagnosis of tiredness foot. The current CNN technique outperforms existing methods and may be employed in the development of future fatigue detection systems [50]. Ahmadi et al. (2021) developed a novel classifier for brain tumor identification based on fuzzy logic and wavelet-based neural networks [51]. Liu et al. (2021) concentrated on issues related to font inpainting. This paper will discuss how to restore broken fonts depending on their style in order to do it more effectively [10]. Rao and Karunakara [52] concentrated on effective segmentation and classification using machine learning models to diagnose tumor development and therapy procedures. Preprocessing, segmentation, extraction, selection, and classification are phases in the suggested approach for efficiently detecting brain tumors. Higher detection accuracy allows for fast and accurate diagnosis, which may save people's lives. As a result, this tumor detection and classification results indicate enhanced performance compared to baseline models. Xu et al. (2020) classified and segmented tongue pictures using a deep neural network. The findings indicate that their technique is extremely compatible with human perception [15].

Maniraj and Maran [53] suggested a hybrid deep learning strategy based on 3D wavelet subband fusion. It is a noninvasive, objective way to examine skin photographs. The provided approach's performance findings on PH2 database photos show that it can efficiently classify normal, benign, and malignant skin images with 99.33 percent average accuracy and more than 90% sensitivity and specificity.

### 3. Methods and Materials

*3.1. An Overview of Deep Convolutional Neural Networks.* Deep learning involves using multiple levels and layers of learning to model high-level abstract concepts using machine learning, artificial intelligence, and other technologies. The idea of deep learning, which was inspired by the natural structure of the human brain and with the help of new facilities and technologies, has been able to achieve considerable success in many areas related to artificial intelligence and machine learning. Recently, artificial intelligence problems can be solved with deep learning algorithms. The most important benefits of deep learning are as follows:

- (i) Automatic features learning
- (ii) Multilayered features learning
- (iii) High accuracy in results
- (iv) High generalization ability and identification of new data
- (v) The potential to create more capabilities and applications in the future
- (vi) Identifying the data features regardless of the location

Deep neural networks, such as convolutional neural networks (CNNs) and ConvNets, are often used for visual and audio processing. Convolutional neural networks use multilayered perceptron types to minimize preprocessing. These are sometimes referred to as shift invariant or space invariant neural networks. Its structure is inspired by the biological processes of the cat's visual cortex such that single neurons only respond to stimulation in a limited area, which is called the reception area. The reception areas of the various neurons overlap in inconsiderable parts to cover the entire field of view. Compared to other data classification approaches, convolutional neural networks use fewer data preprocessing, meaning that the network meets the standard learned manually in previous approaches. This essential advantage is independence from last knowledge and human manipulation in convolutional neural networks. Numerous applications of neural networks have been proposed, including computer vision, suggestive systems, and natural language processing. One of the most significant deep learning techniques is convolutional neural networks, which effectively train many layers. In many applications, this strategy is very common and effective. In general, convolutional neural networks consist of three layers: convolution, pooling, and fully connected layers. Each layer has a distinct function. Feed-forward and backpropagation training are the two stages of convolutional neural networks.

In the first step, the input is provided to the network, and this operation amounts to nothing more than a dot product between the input and the parameters of each neuron. After this, each layer is convolutionalized. The network output is then calculated to determine the network error rate, which is used to adjust the network parameters or, to put it differently, the network training itself. Outputs are compared with a loss function. The backpropagation step is then initiated based on the calculated error rate. In this step, the gradient of each parameter is computed using the chain rules, and each parameter is adjusted according to its influence on the network error. After the settings have been adjusted, the subsequent feeds begin in reverse order. The network's training is complete once a sufficient number of these processes have been completed.

The proposed approach has three main phases, including preprocessing operations for image enhancement, features segmentation and extraction, and a final classification for determining the type of tumor. Features segmentation and extraction operation based on the fractal model are presented in a deep neural network of fuzzy convolution phases and training section, and afterward, the classification occurs. At first, the brain MRI input images must be normalized. Brain MRI images also have a series of noises. Therefore, an initial optimization must be made on these images so that the process occurs in the main sections to achieve a significant result. In the preprocessing phase, input data, which are tumorous and nontumorous images, are normalized and enhanced to increase the efficiency of the system's recognition if necessary.

*3.2. Image Optimization and Preprocessing Operations.* Image optimization and preprocessing operations are performed in the following steps:

*Step 1.* Normalize the image size for changing the image size to a default size such as  $256 \times 256$ . Brain tumor segmentation systems use  $128 \times 128$  or  $256 \times 256$  pixels or the normal collection size based on a unit. The other issue in normalization is called camera distance from the tumor. According to equation (1), all of the measured values relative to this quantity are normalized by an amount named BASE.

$$\text{BASE} = \sqrt{(x_5 - x_{10})^2 + (y_5 - y_{10})^2}. \quad (1)$$

The tumor is considered the reference point to determine the focal length in different images that should be measured relative to a fixed reference, which is assumed to be the origin of face focal length compassion. The coordinate's origin is obtained from the following equations:

$$x_0 = \frac{x_5 - x_{10}}{2}, \quad (2)$$

$$y_0 = \frac{y_5 - y_{10}}{2}. \quad (3)$$

*Step 2.* Conduct histogram integration of the images on images that are dark and should be bright enough to extract important features of the brain tumor, including the intended tumor tissue.

*Step 3.* Intermediate filtering eliminates possible noises.

*Step 4.* High-pass filtering is the process of traits extraction based on brain tumor generalities. It is possible to get better results from edge finding and fragmentation methods. High-pass filtering emphasizes details such as edges, which thereupon increases edge detection efficiency.

*Step 5.* Eliminate the rotational effect obtained from the following equation and also conducted to adjust the image level.

$$\theta = \arctan \frac{y_5 - y_{10}}{x_5 - x_{10}}. \quad (4)$$

In this relation,  $\theta$  is the BASE angle relative to the horizon for eliminating the tumor rotation. The action gives rise to calculating the new coordinates relative to the origin obtained from

$$y_a = y_b - y_{\text{original}}. \quad (5)$$

*3.3. Features Segmentation and Extraction Operation.* The system designer has no control over the operating environment in applications, such as automatic targeting. The usual approach is to focus on selecting the type of sensor that is most likely to increase the desired targets and reduce the share of irrelevant visual details. Most image segmentation

algorithms are based on one of the two basic features of brightness intensity, discontinuity, and similarity. Discontinuity methods divide an image like edges based on sudden changes in brightness intensity. Simulation methods are based on dividing an image into areas similar to each other according to predefined criteria. Thresholding, region growing, cutting, and merging of areas are examples. This study will use fractional-based morphological descriptive-based segmentation descriptors. The morphological operation refers to a branch of biology dealing with shape and structure. The term “mathematical morphology” is used to extract image components useful in expressing features and describing the shape of areas such as boundary areas, frameworks, and convex hulls. The language of mathematical morphology is set theory and a unified and powerful way to deal with image processing issues. In mathematical-based morphology, sets are represented as objects in an image. The fractal model is used for segmentation and extraction operations, which works based on texture, brightness intensity, and edge features. Initially, the durability section identifier is used. For  $A$  and  $V$  sets in  $Z$ , durable areas of  $A$  and  $B$  are displayed as  $A \ominus B$ , defined as

$$A \ominus B = \{z | (B)_z \subseteq A\}. \quad (6)$$

It indicates that the erosion of  $A$  by  $B$  is the set of all points  $z$  such that  $B$  is contained in  $A$  that the components do not share between  $B$  and the background, and the durability regions can be represented as

$$A \ominus B = \{z | (B)_z \cap A^c = \emptyset\}. \quad (7)$$

According to (7),  $A^c$  is the complement of set  $A$  and  $\emptyset$  is the null set. Then, the self-similar operator is used. If  $A$  and  $B$  are set in  $Z^2$ , the dilation of  $A$  by  $B$ , denoted  $A \boxplus B$ , will be displayed in the form

$$A \boxplus B = \{z | (\hat{B})_z \cap A \neq \emptyset\}. \quad (8)$$

In (8),  $A \boxplus B$  is the reflection of  $B$  around its origin and shifting this reflection by  $z$ . Therefore, the dilation of  $A$  by  $B$  is the set of all  $z$  displacements. According to the interpretation, the expanded relationship will be in the form

$$A \boxplus B = \{z | [(\hat{B})_z \cap A] \subseteq A\}. \quad (9)$$

According to (9),  $B$  is a structural component, and  $A$  is a set of image objects on which self-similarity should be conducted, used for the segmentation of boundary extraction in the fractal model. The boundary of a set  $A$  shown with  $\beta(A)$  can be found first with the durable operator  $A$  with  $B$  and then subtracted between set  $A$  and its durability, which is in the form

$$\beta(A) = A - (A \ominus B). \quad (10)$$

According to (10),  $B$  is a suitable structural component. After the image segmentation is done, the feature extraction step will be performed. Brain tumor features extraction from MRI images is a vital process that significantly impacts classification results. The proposed method is obtained by extracting the necessary information from the data.

Therefore, only intrinsic features are selected for data classification. Also, the fractal model has been used to extract the feature. The feature selection has been used in order to deal with high input features and also to reduce the dimensions and identify the most relevant features causing the sufficient separation of different classes. The feature selection method is a sensitive one; thus, insufficient features reduce the classification efficiency, while a larger set of features does not always give better accurate identification results.

Using covariance analysis, fractal features from the image were extracted, generating eigenvalues and minimizing the dimensions. The input images for the fractal algorithm must be of the same size. The two-dimensional matrix of a single image is referred to as a single vector. The first step in the fractal algorithm is to upload training images. These should be grayscale images with a certain resolution. An image containing both the background and the tumor will not be able to provide an accurate diagnosis of the tumor. Each image is converted to a column vector by adding lines, and the images are loaded from a matrix of size  $M \times N$ , where  $N$  is the number of pixels and  $M$  is the number of images. It is necessary to compute the average image to obtain the standard deviation for each original image. We then construct the covariance matrix and obtain the eigenvalues and eigenvectors of the covariance matrix that correspond to the tumor's values. Light intensity, edges, and texture are among the primary features of this research. The number of training images  $M$ , the average of the images  $F_i$ , and the number of images  $L_i$  denote the number of images in the vector  $T_i$ . Initially, there are  $M$  images, each of which contains the  $N \times N$  dimension. Equation (12) describes the process of averaging an image in a three-dimensional  $N$  space.

$$A = N \times N \times M, \quad (11)$$

$$F_i = \frac{1}{M} \sum_{t=1}^m T_t. \quad (12)$$

Finding the standard deviation is an important issue in the fractal algorithm, which is also calculated through equation (13) and the covariance matrix from (14).

$$\text{Variance} = \frac{1}{M} \sum_{t=1}^m T_t, \quad (13)$$

$$\text{Cov} = AA^T. \quad (14)$$

$A = [\text{Variance}_1, \text{Variance}_2, \dots, \text{Variance}_n]$  and  $\text{Cov} = N^2 * N^2$  are a matrix because  $A = N^2 * M$  is a matrix. So  $\text{Cov}$  is huge. Eigenvalues are now obtained from  $\text{Cov}$  using

$$U_i = AV_i. \quad (15)$$

The last step is to choose eigenvectors. A collection of intrinsic state function attributes in the form of  $N$  ( $N = 213$ ) samples in a  $D$ -dimensional space  $\{x_1, x_2, \dots, x_N\}$  is provided.  $x_i = R^d$  and belongs to the  $C$  ( $C = 7$ ) class from

$\{L_i | i = 1, 2, \dots, C\}$ . The fractal algorithm aims to find a linear transmission that maps the original  $D$ -dimensional space to the  $F$ -dimensional space denoted by  $f < d$ . At  $y_i = R\hat{F}$ , the new feature vector is situated. Total scatter or covariance matrices are used to describe scattered matrices in the class. They are computed using (16) and (17).

$$S_T = \sum_{k=1}^N (x_k - \mu)(x_k - \mu)^T, \quad (16)$$

$$W_{\text{Fractal}} = \arg \max [W^T S_T W] = [w_1 w_2 \dots w_f]. \quad (17)$$

$\mu$  is the samples mean and  $\{w_i | i = 1, 2, \dots, f\}$  is a set of special  $S_T$   $F$ -dimensional vectors that are associated with the largest  $f$  eigenvalues. Samples in the new space are  $y = W^T x$ , which is  $W_{\text{Fractal}} \in R^{\text{fxd}} (170 \times d)$ . The main components of the tumor ( $s$ ) are calculated in the training set. Identification operations are formed by tumor designs components in the image space. A comparison is made based on the Euclidean distance of eigenvectors from the main components of the image. The tumor in the image can be identified if the distance is small. On the other hand, if the distance is so large, the image is considered one of the accessories for an independent sample that the system has trained. As a starting point, the training images are read in  $N \times N$  dimensions and converted to  $N^2 \times 1$  dimension.

**3.4. Training and Testing with Deep Fuzzy Convolutional Neural Network Algorithm.** An  $N^2 \times M$  training set is also built-in, and  $M$  is the number of image samples. The average image set is calculated through

$$\psi = \frac{1}{M} \sum_{i=1}^M \Gamma_i. \quad (18)$$

According to (18),  $\psi$  is the average image,  $M$  is the number of images, and  $\Gamma_i$  is the image vector. The corresponding components are maintained with the eigenvalues. These components define the tumor space. The eigenspace is constructed by dragging the image into the tumor space, which results in the formation of the components. As a result, weight vectors are computed. The image dimensions are resized to conform to the standards, and the image is enhanced during the preprocessing stage. The image weight vector is then compared to the tumor weight vector in the image database. The average tumor size is determined and then subtracted from each image in the training set. The result of the subtraction operation is used to form a matrix. The difference between each image and the average image is computed using the formula

$$\phi_i = \Gamma_i - \Psi, \quad i = 1.2. \dots, M. \quad (19)$$

In (19), the difference between the image and the average image is  $\phi$ . The matrix acquired by subtraction is identical to matrix  $A$ ; it is multiplied by its transpose, and lastly, the covariance matrix  $C$  is generated, whose relationship is expressed by

$$C = A^T A. \quad (20)$$

According to (20),  $A$  is formed by the disparity between the vectors; for instance,  $A = [\phi_1, \phi_2, \phi_3, \dots, \phi_M]$ . Dimensions of  $C$  matrix are  $N \times N$ . The number of image samples,  $M$ , is used to form matrix  $C$ . In practice, matrix  $C$  is the same as  $N \times M$ . Alternatively, when the  $A$  order is equal to  $M$ , only  $M$  of  $N$  is the number of eigenvectors equal to the nonzero value. A covariance matrix is then calculated based on the specific values. Several different images are used to segment the components, and the number of eigenvector classes is subtracted. The number of eigenvector classes measures tumors in the image. To make the feature smaller, the eigenvectors are multiplied by matrix  $A$ . As the eigenvectors get smaller, the covariance matrix changes less. However, other features remain the same. Eigenvectors are determined by the accuracy of the images in the image database. Components are groups of eigenvectors. After the components have been obtained, the images in the image database are gathered into the component space, and the image weights are in the same projected image space. An eigencoefficient of a database image is used to determine an image. This is then used to create the component. Calculate the Euclidean distance between the image component and the components collected in the previous step. The tumor is identified as an object whose Euclidean distance is less than the threshold value at the component database. In the case of all Euclidean distances exceeding the threshold, the tumor in the image will not be detected, and the image will be discounted. The convolutional neural network is used as a deep learning technique to improve the results of fragmentation and extraction of the most optimal features in the classification phase. Intending to apply this network, it is necessary to determine the twists and turns, which have three general methods: threshold coefficients wave, adaptive filters, and threshold action potentials scope. This research approach for edge-based segmentation is to use the threshold of the scope of action potentials. The value of this threshold is determined as

$$\left\{ \sigma_n = \text{median} \left\{ \frac{|x|}{0.6745} \right\} \right. \text{Threshold} = 3.5\sigma_n. \quad (21)$$

In equation (21),  $x$  is the signal recorded by the microelectrode (raw signal) and  $\sigma_n$  is an estimated standard noise deviation. It is important to note that a larger value will be obtained for the threshold by using the standard signal deviation. As a result, some rotations will be mistakenly removed. After selecting the threshold value, the rotations are aligned based on their maximum values. Precise alignment of torsions is a very important and decisive factor in edge-based segmentation with a twist. This neural network needs to be trained. The purpose of this training is to find mapping such as  $f: R^n \rightarrow R$  as

$$f(v) = \sum_i^n w_i \varphi(\|v - C_i\|). \quad (22)$$

According to (22),  $v \in R^n$  is a 32-point vector for input, and the Gaussian  $\varphi(0)$  base function is defined as

$$\varphi(v) = \exp\left(\frac{-v^2}{2\sigma^2}\right). \quad (23)$$

Then, for each instruction sample's random initial weight values, the corresponding error with each instance is calculated from the gradient descent as

$$e_i = t_i - y_i = t_i \sum_{j=1}^N w_j \varphi(\|v_i - C_j\|). \quad (24)$$

Therefore, the total error of the network for each training input vector  $P$  of the visual data is equal to  $E = 1/2 \sum_{i=1}^P |e_i|^2$ . If the  $E$  error reaches a lower threshold error, the training ends. This value is determined manually at the beginning. Otherwise, the weights are updated using gradient descent. After completing the training phase with the convolutional neural network, the results are assigned to the class belonging to it.

It should be noted that the main model of this neural network is listed for classification in Section 4. However, it is

necessary to determine its layered structure, which includes the input layer (neurons), the hidden layer in which the training operation takes place, and consists of the three inner layers of pooling layers, fully connected layers, and convolve layers. The final test operation is also in the output layer. The difficulty of detecting tumor presence or absence creates a challenge and a search space. In an optimization problem such as tumor detection, the presence or absence of the tumor from the images will be with the  $N_{\text{var}}$  dimension one array  $X_{n_{\text{var}}}$ , which indicates the current position for its convolve layer in the convolutional neural network. This arrangement is defined as

$$\text{Convolve} = [x_1, x_2, \dots, x_{N_{\text{var}}}], \quad (25)$$

The degree of suitability (or profit value) in the current torsion layer or Convolve is obtained by evaluating the function of the tumor ( $f_p$ ) in Convolve. So there are equations (26) and (27).

$$\text{profit} = f_p(\text{Convolve}) = f_p(x_1, x_2, \dots, x_{N_{\text{var}}}), \quad (26)$$

$$j(x^{(i)}, \dots, x^{(n)}, \theta^{(1)}, \dots, \theta^{(n)}) = \frac{1}{2} \sum_{(i,j):r(i,j)=1} \left( (\theta^{(j)})^T x^{(i)} - y^{(i,j)} \right)^2 + \frac{\lambda}{2} \sum_{i=1}^{n_m} \sum_{k=1}^n (x_k^{(i)})^2 + \frac{\lambda}{2} \sum_{i=1}^{n_u} \sum_{k=1}^n (\theta_k^{(j)})^2. \quad (27)$$

The function of the overall goal in diagnosing tumors or not is in the form of equation (27). In general, we should minimize this function as much as possible. It is a case of removing additional sections to accurately identify the area, which is minimized from

$$j(x^{(1)}, \dots, x^{(n)}, \theta^{(1)}, \dots, \theta^{(n)}). \quad (28)$$

As evidence, the structure of the deep convolutional neural network used is an algorithm that maximizes the function of detecting a tumor or not. Using the deep neural network convolution to solve minimization problems such as tumor diagnosis only needs to multiply a minus by the cost function. To start the convolution optimization algorithm, a convolve matrix is generated to the size  $N_{\text{pop}} * N_{\text{var}}$ . Then, a random number of pooling layers is assigned for each of these convolutions. Basically, the pooling layers are between 2 and 5 items. These numbers are used as the upper and lower limits of the pooling allocation to each twisting section in the deep training in different iterations. Another deep convolutional neural network pattern is that they have connected layers at a certain range. Hence, the maximum amplitude of the connected layers in the neural network is called  $\text{Max}_{\text{ConnectedLayer}}$ . In an optimization problem, the variable higher limit  $\text{var}_{\text{high}}$  and the lower limit  $\text{var}_{\text{low}}$  will each have a deep layer of  $\text{Max}_{\text{ConnectedLayer}}$ , which corresponds to the total layers number, current layers number,

and also high and low variables. Therefore,  $\text{Max}_{\text{ConnectedLayer}}$  is defined as

$$\text{Max}_{\text{ConnectedLayer}} = \alpha \times \frac{\text{number of current layers}}{\text{total number of layers}} \times (\text{Var}_{\text{high}} - \text{Var}_{\text{low}}). \quad (29)$$

In (29),  $\alpha$  is the variable with which the maximum value of  $\text{Max}_{\text{ConnectedLayer}}$  is set. In (28),  $\theta$  is the layers, and in equation (27),  $\lambda$  is the evaluator value. Each torsional section in the deep neural network of the convolution traverses only 1% of the total detected areas toward the current ideal target and has  $\varphi$  radian deviation. These two parameters help each torsion section in the deep convolutional neural network to explore more of the space.  $\lambda$  is a random number between 0 and 1, and  $\varphi$  is a number between  $\pi/6$  and  $-\pi/6$ . When all the torsion sections in the deep convolutional neural network migrate to the target point and new habitat points are identified, each torsion section in the deep convolutional neural network acquires a number of layers of communication. Depending on the number of layers in each torsion section in the deep neural network, a  $\text{Max}_{\text{ConnectedLayer}}$  is determined for it. Now it is necessary to fuzzy this deep neural network of convolution. It is assumed that the image data set is in the form of  $M$ , which is indicated the number of



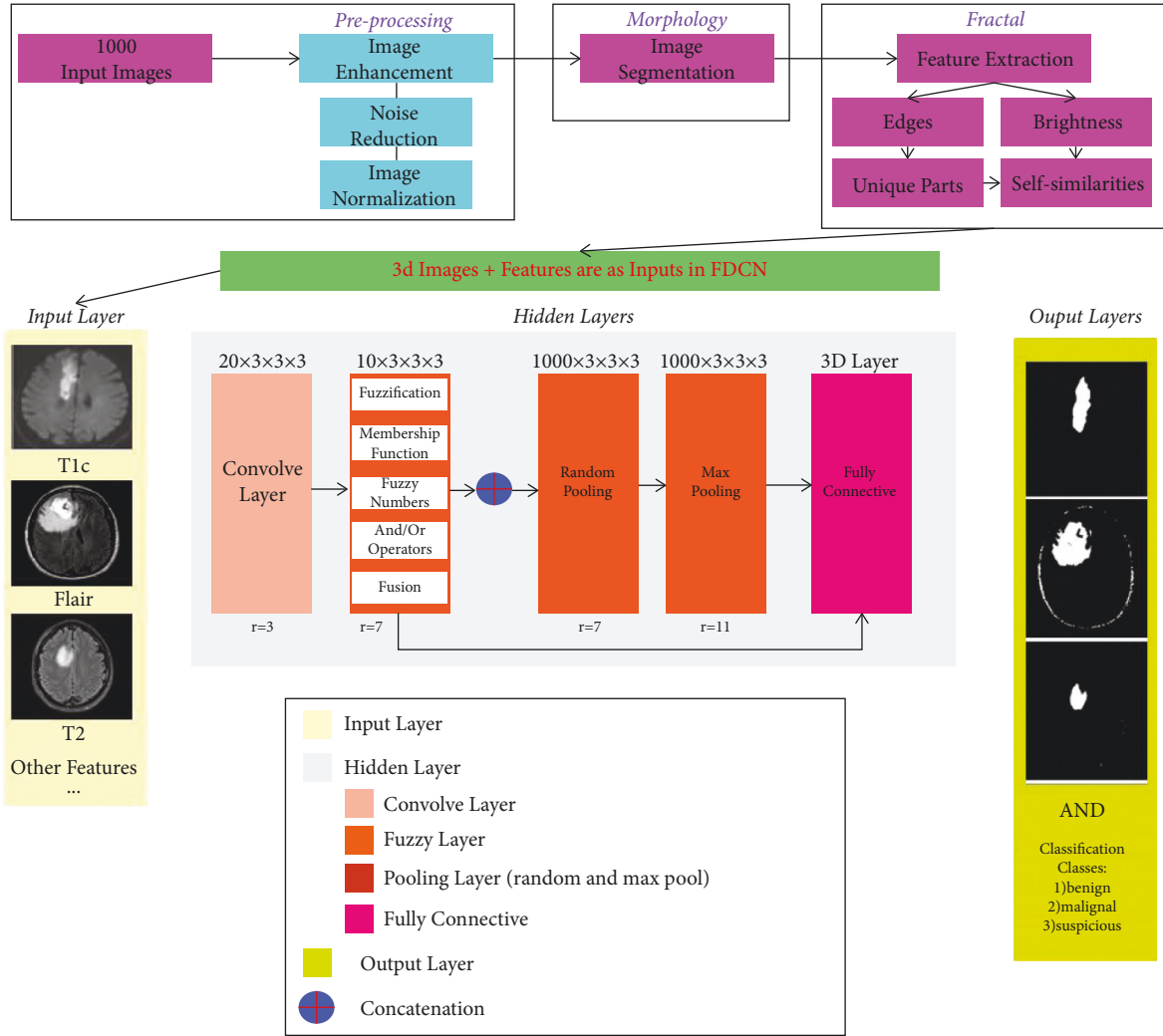


FIGURE 1: The proposed approach flowchart.

training images, the average  $F_i$  of the images, and the  $L_i$  of each image of the  $T_i$  vector. Initially,  $M$  has a number of images, each containing the  $M \times N$  dimension. Each image can be displayed in  $N$ -dimensional space as  $A = N * N * M$ . Therefore, it is necessary to provide a combination layer for the fuzzy part with a deep convolutional neural network, which will be in the form of

$$x_i^{(l+1)} = (w_d)_i^{(l+1)} (y_d)^{(l)} + (w_f)_i^{(l+1)} (y_f)^l + b_i^{(l+1)}. \quad (30)$$

In (30),  $y_d$  is the output of the deep convolutional neural network section and  $y_f$  is the output of the fuzzy section, which has two sections for weight,  $w_d$  and  $w_f$ .  $w_d$  determines the weight of the deep convolutional neural network and  $w_f$  determines the weight of the fuzzy part. This layer is used as  $\hat{y}$  to combine the deep neural network of the convolution and the fuzzy part, which has a transition function that is considered to be the hyperbolic tangent. In fact, the classification and diagnosis section was presented using an innovative method called fuzzy deep convolutional neural network or FDCNet. What separates the proposed approach from the structure of ANFIS (Neurofuzzy)

networks is that, in ANFIS (Neurofuzzy) network structures, the definition of membership functions and fuzzy numbers is based on inputs and some attributes (especially in images where the attributes are considered) and is connected to an extension system that is trained based on a weighted training layer and bias with a transfer or stimulus function. But in the proposed approach structure that performs the detection and classification operation, the fuzzy structure is located on the training layers or hidden layers of the deep neural network section of the convolution after the torsion layer. The filter applied on the torsion layer, the fuzzy layer, the pooling layer, and the completely connected layer is in the form of a  $3 \times 3$  window.

In fact, in this training operation, the interfuzzy coefficients are applied with the layers of convolution to wit the torsion layer, and the output layer displays the classification based on image features as a vector matrix. Definitive processing is performed on the input image to prove the fuzzy uncertainty problem. Also, fragmentation and features extraction based on morphological processing and fractal model uses a three-dimensional matrix in the form of ground truth to accurately and carefully identify features,

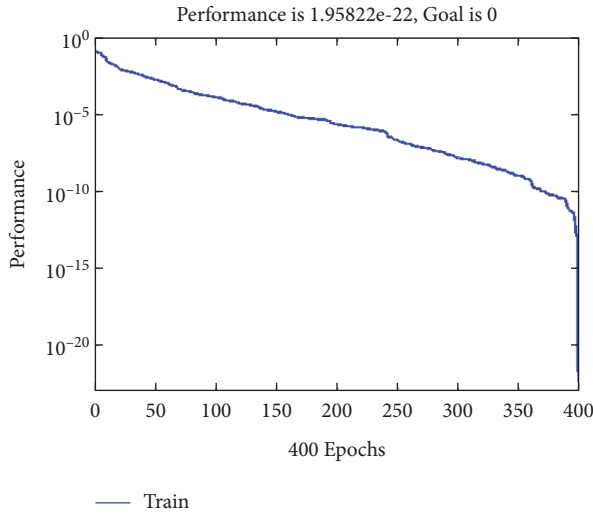


FIGURE 2: Performance diagram to achieve convergence and end goal in the training phase.

then reduce the dimensions, and select and extract the features.

**3.5. The Proposed Approach Diagrams.** The proposed approach presented in this study has a general trend in Figure 1. This diagram shows that the preprocessing operation starts with 1000 three-dimensional input images, which are from the BraTS data collection and are in the form of DICOM images. Then, the images are improved with the middle filter method to reduce noise and unify their size. Then, the image fragmentation operation based on the intensity of light and edges occurs, which is morphology-based; according to this fragmentation, durable and self-similarity operators of the fractal method begin to work. They can reduce the dimensions, choose the best features, and perform features extraction with high accuracy. The training and testing operation then begins with a deep fuzzy convolutional neural network or a method called FDCNet. This neural network is different from neural-fuzzy networks (ANFIS) in structure because its fuzzy part is considered in the hidden layer in the training section. In contrast, the fuzzy part is first executed in the neural-fuzzy network. Training operations are performed based on input, membership functions, fuzzy numbers, and the rules defined in it.

The FDCNet method inputs, in which the neurons are located, are the three-dimensional images of the data and their features. The training operation is supposed to be performed on three-dimensional images to determine tumors, but in the final classification, in addition to the original image, features are needed to create benign, malignant, and suspicious classes. The FDCNet structure, after reading the data from the input layer, is to create a hidden layer including 20-channel twist layers and a  $3 \times 3$  filter, which is  $3 \times 3 \times 3$  in three-dimensional images and has an Astros rate of 3 ( $r=3$ ). Next is the fuzzy layer, along which fuzzy data training in the twisted layer determines membership functions, fuzzy numbers, And/Or rules, and then

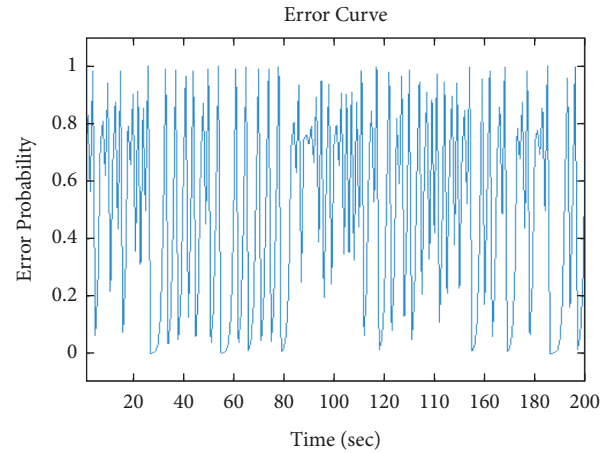


FIGURE 3: The error graph at the time of training from the whole data set is based on an error probability at the simulation time.

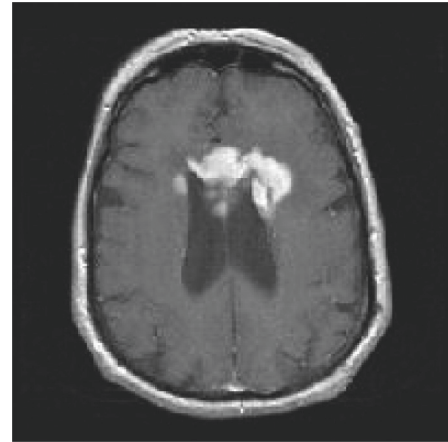


FIGURE 4: Input image.

fusion. Also, its Astros rate is 7 ( $r=7$ ), and the filter applied to it is  $3 \times 3.3$ . There are also 10 layers in this fuzzy section. Then the training and classification operation in the pooling layer is applied in two parts: random pooling and max pooling, which has 1000 data equal to the layers for training (i.e., 100% of the data are trained, and any image from the data set can be tested on). The filter applied to it is  $3 \times 3$  with an Astros rate of 11. Then there is the 3-dimensional layer of the data test as a completely connected layer, which is also affected by the fuzzy layer. The data tests performed in this layer transfer to the output layer and display the tumor area visually and then create classes to display the type of tumor as malignant, benign, and suspicious.

## 4. Results and Discussion

The set of brain tumor data used in this study is BraTS data. It has 145 files for individuals undergoing various MRI imaging procedures. This data collection has four versions ranging from 2012 to 2018, with each version increasing the quantity of data pieces and their quality. The primary data is in DICOM format, converted to JPEG for better usage with

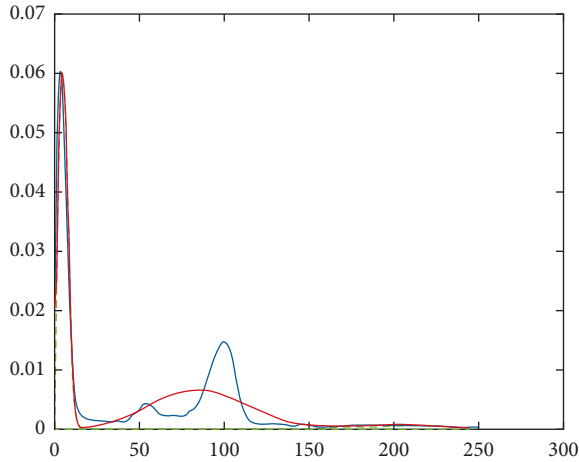


FIGURE 5: Histogram and color channel separation of the input image.



FIGURE 6: Converted image from RGB to gray surface.



FIGURE 7: The result of image fragmentation based on the fractal model.

the DICOM Viewer program in this study. Three-dimensional graphics are used as input. Due to the vast image size in the BraTS data set, this study employs 1000 video input samples to validate the suggested technique. Several images have been used that will be displayed in the results section. The simulation takes place in MATLAB 2015b and will run on a system with a 7-core processor with 6 Mbps cache and 3.6 MHz and 6 GB of memory in Windows 8. First of all, the



FIGURE 8: The result of final fragmentation in the neural network training phases of deep fuzzy convolution.

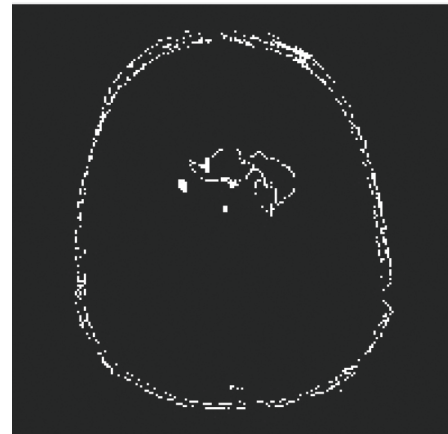


FIGURE 9: The final result of the proposed method.

structure of the deep neural network of fuzzy convolution should be considered, and the number of layers and the number of neurons should be determined. Initially, 32 primary neurons are considered in the input layer, which includes all the features of BraTS data. There are settings in the hidden layer with three main sections in the neural network, deep fuzzy convolution depth, namely, twisting, pooling, and completely connected. The sum of these layers is 4 cases creating a  $3 \times 3$  torsional matrix. This means that, in each of the 4 layers, there is also a  $3 \times 3$  matrix. The torsion layer is a single layer, and the pooling layer consists of two layers, one part of which is considered maximal or the so-called *maxpool*, and the other part is random pooling that can teach each of the features at random. So there is one torsion layer, two pooling layers, and one fully connected layer, and the output layer contains any feature based on distinguishing features from MRI images. These layers are also connected to a fuzzy structure. Training section layers are arranged in the following order: a twisting layer, a deep fuzzy layer with membership functions that are calculated from the output features, the original three-dimensional images, two layers of pooling layer including the random part and the *maxpool* section, and the fully connected layer. There is a problem called centroid, which is considered in the

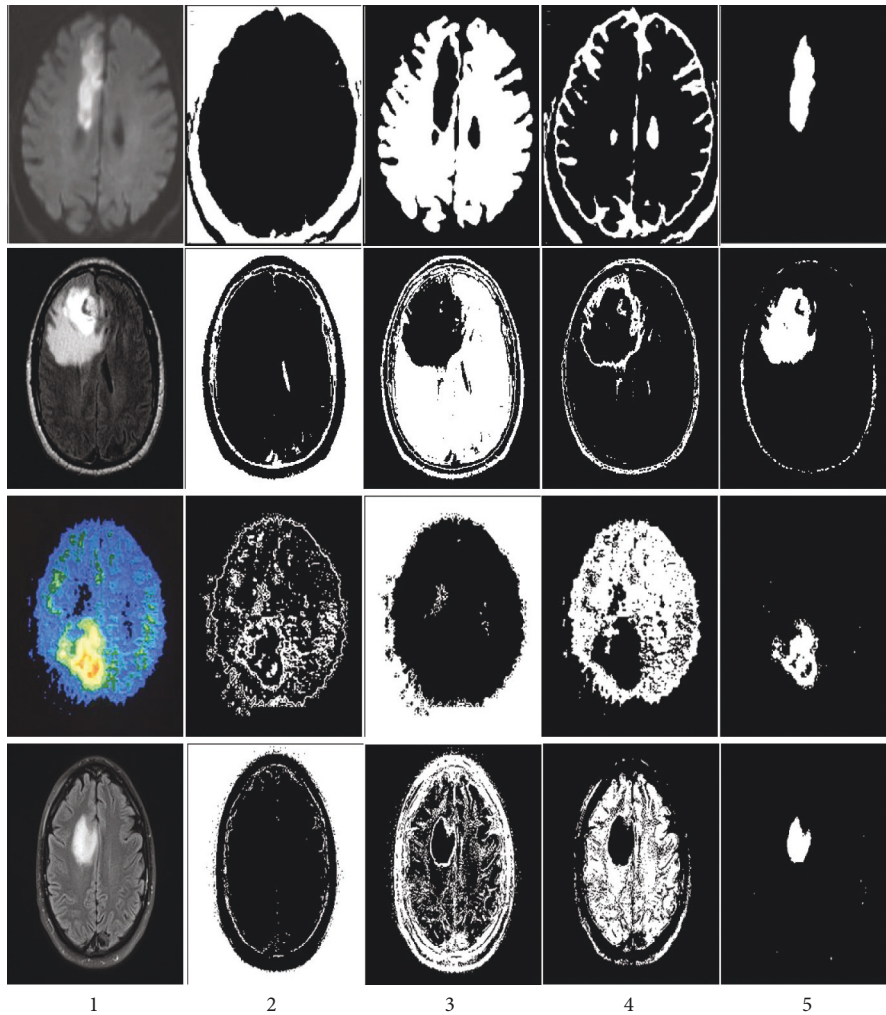


FIGURE 10: The results of the proposed method on other images.

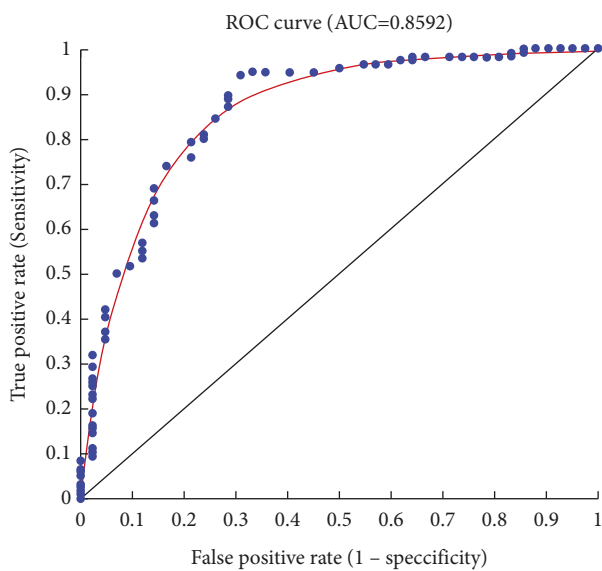


FIGURE 11: ROC diagram and AUC rate.

TABLE 1: Evaluation results.

Parameter	PR
Accuracy	98.6891%
Sensitivity	94.8766
Mean square error	0.8598
Peak signal-to-noise ratio	21,2646 dB
Signal-to-noise ratio	26,8642 dB
Area under curve	0.8592

principles of classification and even clustering to perform detection and work tracking. The structure is individual, that is,  $3 \times 3$ ,  $5 \times 5$ , and  $7 \times 7$ . A house or pixel is placed in the middle, the adjacent houses are analyzed, and that central pixel is considered the centroid. It is filtering on the layers of the training or hidden section measured as  $3 \times 3$  in small dimensions. The results and simulation outputs performed will be placed in this section. First, all images will be read. The operation begins in the training and testing phase with the neural convolutional network and features extraction with the fractal model.

TABLE 2: The results of the proposed method with other methods.

Reference	Proposed method	Precision (%)
Chen et al., 2019 [3]	Using generative adversarial networks (GAN)	94.5
Han et al., 2019–2020 [4]	PGGAN	95
Özyurt et al., 2019 [5]	Convolutional neural network with a combinational method called NS-CNN-EMFSE	96.52
Sajjad et al., 2019 [9]	Deep convolutional neural network with expanded data amplification	95.5
Özyurt et al., 2020 [13]	Using a super-pixel image with C-means fuzzy combinational method and convolutional neural network	98.33
Amin et al., 2020 [54]	Discrete wavelet transform method based on convolutional neural network (DWT-CNN)	98
Ahmadi et al., 2020 [14]	Optimization of quantum matched-filter techniques and deep spiking neural Networks (QAIS-DSNN methods)	91.92
The proposed method	The fractal segmentation method is based on a deep fuzzy convolutional neural network or FDCNet	98.68

It should be noted that the three-dimensional MRI image will be processed. Then for the final classification, the features of the fragmentation are used, which are a combination based on morphological processing and a fractal model that creates a ground truth three-dimensional matrix. Figure 2 shows the performance chart to achieve convergence and the ultimate goal in the training phase. To test the data, an input image needs to be considered. 70% of the data are trained, and 30% are tested. But one of the images needs to be shown visually (see Figure 3). We used one of the images in the data set, shown in Figure 4. Then, the histogram and the separation of the color channels of that image are calculated, which is similar to Figure 5. This histogram and color channel separation show the red, green, and blue color channels based on 256 colors on a threshold. The image is then normalized and converted from color mode or RGB color space to gray, resulting in Figure 6.

Then segmentation based on the fractional model by local features, including texture, light intensity, and image edge, based on two factors including self-similarity and durability, is as shown in Figure 7. The fragmentation is performed by identifying three features, including light intensity, edges separated from the original image, and texture, which is shown in Figure 8. The used features are based on edge finding (connected points in another object), light intensity, and texture. Three local features are used for fragmentation. In the basic sciences, in comparison to the methods that use more number of features, those methods with lower feature numbers with appropriate answers have higher scientific validity. Finally, the fuzzy deep neural network will be used for the final diagnosis, aiming to improve the segmentation and select the optimal class of features to detect the features in the MRI images in the image, resulting in Figure 9. As shown in Figure 9, the image segmentation and classification operations are well performed, and the area of tumor features is identified. In Figure 10, the method presented is applied to several other images in the data set.

In addition to the features mentioned above, it is essential to consider local characteristics, including light intensity, edges, and any other characteristic that may have been identified during the segmentation and feature extraction steps, as well as tumor classification operations, such as benign, malignant, and suspicious. This is where the

three-dimensional ground truth matrix begins to work using the image properties to consider the classification. This curve measures the system's ability to classify or cluster data. It is a graphical depiction of the degree of sensitivity in a binary classification system with variable separation thresholds or the ratio of true to erroneous predictions. In addition to graphing true positive predictions versus incorrect positive predictions, the receiver operating characteristic (ROC) curve can also be seen. An area under the receiver operating characteristic curve is a numerical value used to quantify and analyze some aspect of performance. Its value varies between 0 and 1. AUC stands for the area under the curve. A value of 0.5 indicates no prediction, whereas values between 0.7 and 1 indicate good predictions, classifications, or groupings. In Figure 11, the AUC value is less than one, suggesting that the suggested technique has been optimized as much as possible. A combination of pieces resembling malignant tumors in the available data and the approach described in this research produced a series of small inaccuracies that are detectable when the fitting line is not matched (Table 1). These blue circles represent the data's qualities, while the red lines represent the data's fit to the ROC diagram. When the data is a little out of reach, errors occur. The middle line also includes regression. The area between the ROC peak and the regression line is the AUC. Table 2 shows the comparison of the proposed method with several other new methods that were studied in previous studies. This comparison is based on the accuracy criterion in terms of percentage. All methods use the same data set, the BraTS.

## 5. Conclusion

Smartification principles in the construction of intelligent medical diagnostic systems to have reliable and rapid procedure have been required. Intelligent medical diagnostic systems could reduce human error and assist physicians in diagnosing. This identification and early detection lead to health recognition and further care until complete recovery. Cancer masses that form in different body parts do not have a regular shape or specific principles. Imaging other body areas can help identify the size and area of these tumors. It is also possible to estimate malignant and benign conditions using medical courses. These tumor masses, which are one of the world's leading death causes, need to be diagnosed as

accurately as possible. Therefore, building intelligent systems is vital and inevitable. In this study, an attempt was made to use brain imaging to produce MRI images that can contain tumors. This study's data set includes BraTS, a series of standard MRI images of the brain surface that have the same features in light intensity, brightness, and color states. Using the principles of image processing and machine learning, a comprehensive system for detecting brain tumors from the level of MRI images can be provided. The proposed approach method initially provided a preprocessing step to reduce noise and smooth image size. In the following, the fragmentation operation based on morphology and, at the same time, the feature extraction operation with the fractal model are presented. In the continuation of the diagnostic and classification operations, the innovative deep neural network of fuzzy convolution or FDCNet has been considered. Its accuracy results have been 98.68.

This method offered a functional advantage over previous methods, including tremendously generative adversarial networks (GAN), progressively growing adversarial networks (PGGAN), convolutional deep neural networks with extensive data enhancement, using super-pixel images with fuzzy C-Means methods, and convolutional neural networks, as well as a discrete wavelet transform method based on convolutional neural networks (DWT-CNN). The existence of segmentation that is comparable to malignant masses in the available data, together with the approach used in this study, has resulted in a series of small inaccuracies that can be seen on the fitting line. An error happens in some regions when the data is out of reach. Based on the results, the presented method's accuracy, sensitivity, and AUC are 98.69%, 94.88%, and 85.92%, respectively. For the future of the study, the presented method should be extended with edge detection methods. Moreover, the optimization for the clustering method can be done using metaheuristic search algorithms such as Genetic Algorithm and particle swarm optimization [55].

## Data Availability

Data are available and can be provided over the emails querying directly to the corresponding author (y.pourasad@uut.ac.ir).

## Conflicts of Interest

The authors declare that they do not have any conflicts of interest

## References

- [1] P. Domingues, M. González-Tablas, Á. Otero et al., "Tumor infiltrating immune cells in gliomas and meningiomas," *Brain, Behavior, and Immunity*, vol. 53, pp. 1–15, 2016.
- [2] Z. Li, Y. Wang, and J. Yu, "Brain tumor segmentation using an adversarial network Brainlesion: Glioma Multiple Sclerosis, Stroke and Traumatic Brain Injuries," in *Proceedings of the International MICCAI Brainlesion Workshop, Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*, pp. 123–132, Quebec, Canada, September 2017.
- [3] H. Chen, Z. Qin, Yi. Ding, and L. Tian, "Brain Tumor Segmentation with Generative Adversarial Nets," in *Proceedings of the 2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD)*, pp. 25–28, Chengdu, China, May 2019.
- [4] C. Han, L. Rundo, R. Araki et al., "Infinite Brain MR Images: PGGAN-Based Data Augmentation for Tumor Detection," in *Neural Approaches to Dynamics of Signal Exchanges*, vol. 151, pp. 291–303, Springer Science and Business Media Deutschland GmbH, Singapore, 2019.
- [5] F. Özyurt, E. Sert, E. Avci, and E. Dogantekin, "Brain tumor detection based on Convolutional Neural Network with neutrosophic expert maximum fuzzy sure entropy," *Measurement*, vol. 147, Article ID 106830, 2019.
- [6] H. Liu, J. Liu, S. Hou, T. Tao, and J. Han, "Perception consistency ultrasound image super-resolution via self-supervised CycleGAN," *Neural Computing & Applications*, 2021.
- [7] M. Zhang, Y. Chen, and W. Susilo, "PPO-CPQ: a privacy-preserving optimization of clinical pathway query for E-healthcare systems," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 10660–10672, 2020.
- [8] W. Tang, S. Wan, Z. Yang, A. E. Teschendorff, Q. Zou, and C. Sahinalp, "Tumor origin detection with tissue-specific miRNA and DNA methylation markers," *Bioinformatics*, vol. 34, no. 3, pp. 398–406, 2018.
- [9] M. Sajjad, S. Khan, K. Muhammad, W. Wu, A. Ullah, and S. W. Baik, "Multi-grade brain tumor classification using deep CNN with extensive data augmentation," *Journal of Computational Science*, vol. 30, pp. 174–182, 2019.
- [10] R. Liu, X. Wang, H. Lu et al., "SCCGAN: style and characters inpainting based on CGAN," *Mobile Networks and Applications*, vol. 26, no. 1, pp. 3–12, 2021.
- [11] J. Chen, Q. Zou, and J. Li, "DeepM6ASeq-EL: prediction of human N6-methyladenosine (m6A) sites with LSTM and ensemble learning," *Frontiers of Computer Science*, vol. 16, no. 2, Article ID 162302, 2021.
- [12] M. Ahmadi, M. Soofiabadi, M. Nikpour, H. Naderi, L. Abdullah, and B. Arandian, "Developing a deep neural network with fuzzy wavelets and integrating an inline PSO to predict energy consumption patterns in urban buildings," *Mathematics*, vol. 10, no. 8, p. 1270, 2022.
- [13] F. Özyurt, E. Sert, and D. Avci, "An expert system for brain tumor detection: fuzzy C-means with super resolution and convolutional neural network with extreme learning machine," *Medical Hypotheses*, vol. 134, Article ID 109433, 2020.
- [14] M. Ahmadi, A. Sharifi, S. Hassantabar, and S. Enayati, "QAIS-DSNN: tumor area segmentation of MRI image with optimized quantum matched-filter technique and deep spiking neural network," *BioMed Research International*, vol. 2021, pp. 1–16, Article ID 6653879, 2021.
- [15] Q. Xu, Y. Zeng, W. Tang et al., "Multi-task Joint learning model for segmenting and classifying tongue images using a deep neural network," *IEEE journal of biomedical and health informatics*, vol. 24, no. 9, pp. 2481–2489, 2020.
- [16] G. R. Chandra and K. R. H. Rao, "Tumor detection in brain using genetic algorithm," *Procedia Computer Science*, vol. 79, pp. 449–457, 2016.
- [17] M. Karnan and K. Selvanayagi, "Improved implementation of brain MR image segmentation using meta heuristic algorithms," in *Proceedings of the 2010 IEEE International Conference on Computational Intelligence and Computing Research*, pp. 1–4, Coimbatore, India, December 2010.
- [18] A. M. N. Darwiesh, N. M. A.-E. Maboud, and A. M. R. Khalil, "Role of magnetic resonance spectroscopy & diffusion

- weighted imaging in differentiation of supratentorial brain tumors,” *The Egyptian Journal of Radiology and Nuclear Medicine*, vol. 47, no. 3, pp. 1037–1042, 2016.
- [19] M. Ahmadi, A. Taghavirashidizadeh, D. Javaheri, A. Masoumian, S. J. Ghoushchi, and Y. Pourasad, “DQRE-SCnet: A Novel Hybrid Approach for Selecting Users in Federated Learning with Deep-Q-Reinforcement Learning Based on Spectral Clustering,” *Journal of King Saud University-Computer and Information Sciences*, 2021.
- [20] E. Eslami and H.-B. Yun, “Attention-based multi-scale convolutional neural network (A+ MCNN) for multi-class classification in road images,” *Sensors*, vol. 21, no. 15, 2021.
- [21] A. Ala, F. E. Alsaadi, M. Ahmadi, and S. Mirjalili, “Optimization of an appointment scheduling problem for healthcare systems based on the quality of fairness service using whale optimization algorithm and NSGA-II,” *Scientific Reports*, vol. 11, no. 1, p. 19816, 2021.
- [22] M. Q. H. Abadi, S. Rahmati, A. Sharifi, and M. Ahmadi, “HSSAGA: designation and scheduling of nurses for taking care of COVID-19 patients using novel method of hybrid salp swarm algorithm and genetic algorithm,” *Applied Soft Computing*, vol. 108, Article ID 107449, 2021.
- [23] M. Mahmoudi, “COVID lessons: was there any way to reduce the negative effect of COVID-19 on the United States economy?” 2022, <https://arxiv.org/abs/2201.00274>.
- [24] E. Sadeghipour, S. Nasrollah, M.-R. Sayebani, and R. Mahdizadeh, “Breast cancer detection based on a hybrid approach of firefly algorithm and intelligent systems,” *Indian Journal of Fundamental and Applied Life Sciences*, vol. 5, p. S1, 2015.
- [25] M. Ahmadi, A. Sharifi, M. Jafarian Fard, and N. Soleimani, “Detection of brain lesion location in MRI images using convolutional neural network and robust PCA,” *International Journal of Neuroscience*, pp. 1–12, 2021.
- [26] N. M. Yazdani, M. Shariat Panahi, and E. S. Poor, “Intelligent detection of intrusion into databases using extended classifier system,” *International Journal of Electrical & Computer Engineering*, vol. 3, no. 5, 2013.
- [27] M. Rezaei, F. Farahanipad, A. Dillhoff, R. Elmasri, and V. Athitsos, “Weakly-supervised hand part segmentation from depth images,” in *Proceedings of the 14th Pervasive Technologies Related to Assistive Environments Conference*, pp. 218–225, Corfu, Greece, 2021.
- [28] L. Mobasheri, M. Hossein Nasirpour, E. Masoumi, A. Foolady Azarnaminy, M. Jafari, and S.-A. Esmaeili, “SARS-CoV-2 triggering autoimmune diseases,” *Cytokine*, Article ID 155873, 2022.
- [29] S. Hassantabar, M. Ahmadi, and A. Sharifi, “Diagnosis and detection of infected tissue of COVID-19 patients based on lung X-ray image using convolutional neural network approaches,” *Chaos, Solitons & Fractals*, vol. 140, Article ID 110170, 2020.
- [30] S. Dorosti, S. Jafarzadeh Ghoushchi, E. Sobhrakhshankhah, M. Ahmadi, and A. Sharifi, “Application of gene expression programming and sensitivity analyses in analyzing effective parameters in gastric cancer tumor size and location,” *Soft Computing*, vol. 24, no. 13, pp. 9943–9964, 2020.
- [31] Y. Yan, L. Feng, M. Shi, C. Cui, and Y. Liu, “Effect of plasma-activated water on the structure and in vitro digestibility of waxy and normal maize starches during heat-moisture treatment,” *Food Chemistry*, vol. 306, Article ID 125589, 2020.
- [32] M. Shi, F. Wang, P. Lan et al., “Effect of ultrasonic intensity on structure and properties of wheat starch-monoglyceride complex and its influence on quality of norther-style Chinese steamed bread,” *Food Science and Technology*, vol. 138, Article ID 110677, 2021.
- [33] R. Yang, M. Xu, T. Liu, Z. Wang, and Z. Guan, “Enhancing quality for HEVC compressed videos,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 7, pp. 2039–2054, 2019.
- [34] A. Aslam, E. Khan, and M. S. Beg, “Improved edge detection algorithm for brain tumor segmentation,” *Procedia Computer Science*, vol. 58, pp. 430–437, 2015.
- [35] R. Rajeswari and G. Gunasekaran, “Tumor detection and segmentation using watershed and hierarchical clustering algorithms,” *International*, vol. 2, 2014.
- [36] Journal of Innovative Research, *Computer and Communication Engineering*, vol. 2, no. 5, 2014.
- [37] M. Xu, C. Li, Z. Chen, Z. Wang, and Z. Guan, “Assessing visual quality of omnidirectional videos,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 12, pp. 3516–3530, 2019.
- [38] H. Sheng, S. Wang, Y. Zhang et al., “Near-online tracking with Co-occurrence Constraints in blockchain-based edge computing,” *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2193–2207, 2021.
- [39] Q. Ain, M. A. Jaffar, and T. S. Choi, “Fuzzy anisotropic diffusion based segmentation and texture based ensemble classification of brain tumor,” *Applied Soft Computing*, vol. 21, pp. 330–340, 2014.
- [40] G. Vishnuvarthanan, M. Pallikonda Rajasekaran, P. Subbaraj, and A. Vishnuvarthanan, “An unsupervised learning method with a clustering approach for tumor identification and tissue segmentation in magnetic resonance brain images,” *Applied Soft Computing*, vol. 38, 2015.
- [41] L. Chen, M. He, M. Zhang et al., “The Role of non-coding RNAs in colorectal cancer, with a focus on its autophagy,” *Pharmacology & Therapeutics*, vol. 226, Article ID 107868, 2021.
- [42] Y. He, L. Dai, and H. Zhang, “Multi-branch deep residual learning for clustering and beamforming in user-centric network,” *IEEE Communications Letters*, vol. 24, no. 10, pp. 2221–2225, 2020.
- [43] A. Ortiz, J. M. Gorriz, J. Ramirez, and D. Salas-Gonzalez, “Improving MR brain image segmentation using self-organising maps and entropy-gradient clustering,” *Information Sciences*, vol. 262, pp. 117–136, 2014.
- [44] N. Nabizadeh and M. Kubat, “Brain tumors detection and segmentation in MR images: Gabor wavelet vs. statistical features,” *Computers & Electrical Engineering*, vol. 45, pp. 286–301, 2015.
- [45] P. Shanthakumar and P. Ganeshkumar, “Performance analysis of classifier for brain tumor detection and diagnosis,” *Computers & Electrical Engineering*, vol. 45, pp. 302–311, 2015.
- [46] M. Niu, Y. Lin, and Q. Zou, “sgRNACNN: identifying sgRNA on-target activity in four crops using ensembles of convolutional neural networks,” *Plant Molecular Biology*, vol. 105, no. 4–5, pp. 483–495, 2021.
- [47] S. Sun, L. Xu, Q. Zou, G. Wang, and J. Gorodkin, “BP4RNAseq: a babysitter package for retrospective and newly generated RNA-seq data analyses using both alignment-based and alignment-free quantification method,” *Bioinformatics*, vol. 37, no. 9, pp. 1319–1321, 2021.
- [48] T. A. Olowookere, “Hybridized Deep Convolutional Neural Network and Fuzzy Support Vector Machines for Breast Cancer Detection,” *SN COMPUT. SCI.*, vol. 3, 2022.

- [49] M. Ragab, A. Albukhari, J. Alyami, and R. F. Mansour, "Ensemble deep-learning-enabled clinical decision support system for breast cancer diagnosis and classification on ultrasound images," *Biology*, vol. 11, no. 3, p. 439, 2022.
- [50] A. Sharifi, M. Ahmadi, M. A. Mehni, S. Jafarzadeh Ghoushchi, and Y. Pourasad, "Experimental and numerical diagnosis of fatigue foot using convolutional neural network," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 24, no. 16, pp. 1828–1840, 2021.
- [51] M. Ahmadi, F. Dashti Ahangar, N. Astaraki, M. Abbasi, and B. Babaei, "FWNNet: Presentation of a New Classifier of Brain Tumor Diagnosis Based on Fuzzy Logic and the Wavelet-Based Neural Network Using Machine-Learning Methods," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 8542637, 13 pages, 2021.
- [52] C. S. Rao and K. Karunakara, "Efficient detection and classification of brain tumor using kernel based SVM for MRI," *Multimedia Tools and Applications*, vol. 81, pp. 1–25, 2022.
- [53] S. P. Maniraj and P. S. Maran, "A hybrid deep learning approach for skin cancer diagnosis using subband fusion of 3D wavelets," *The Journal of Supercomputing*, vol. 73, pp. 1–16, 2022.
- [54] J. Amin, M. Sharif, N. Gul, M. Yasmin, and S. Ali Shad, "Brain tumor classification based on DWT fusion of MRI sequences using convolutional neural network," *Pattern Recognition Letters*, vol. 129, pp. 115–122, 2020.
- [55] R. Farzaneh, A. Malek, F. Mirzaei, S. Amiri, F. Salehpour, and A. Z. S. A. F. Meshkini, "Evaluation of comorbid psychiatric disorders in patients with primary brain tumors before and after surgery," *Neurology Psychiatry and Brain Research*, vol. 36, pp. 8–13, 2020.



## Research Article

# The Method for Identifying Employees' Emotions in Adverse States Incorporating PSO-kNN Algorithm and Multiple Physiological Parameters

Jiaonan Han 

Human Resources Business Consulting Department, Kunlun Digital Technology Co. Ltd., Beijing 100007, China

Correspondence should be addressed to Jiaonan Han; hanjiaonan@cnpc.com.cn

Received 16 February 2022; Revised 3 March 2022; Accepted 5 April 2022; Published 23 April 2022

Academic Editor: Mohamed Abdelaziz

Copyright © 2022 Jiaonan Han. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

It is well known that we, as human beings, are prone to a variety of undesirable emotions such as excitement, boredom, and fear, all of which are induced by varying degrees of negative states. In this paper, we designed an emotion-evoking experiment to induce calm, excited, bored, and fearful emotions, as well as low, moderate, and high levels of tension. Based on the six physiological signals such as heart rate and respiration rate of the subjects in these emotion states, feature extraction was performed after removing the baseline preprocessing, combined with particle swarm optimisation algorithm for feature selection, and the k-nearest neighbour algorithm was used to classify the different emotion and tension levels in the undesirable states. By comparing the results of several sets of experiments, we found that with baseline removal and particle swarm feature selection optimisation, our experimental results using k-nearest neighbour classification showed a significant improvement in recognition accuracy compared to the traditional k-nearest neighbour algorithm, which indicates that the proposed method has better recognition results.

## 1. Introduction

Emotions are a combination of states that arise when a person is exposed to external stimuli. A good emotional state is conducive to maintaining physical and mental health, while chronic bad moods can have a great impact on a person's mental health and physical health. For example, prolonged bad moods can easily lead to depression, which affects one's social functioning and interpersonal interactions and can even be life-threatening [1]. For people with cardiovascular diseases, extreme emotions such as anger and anxiety can increase the risk of morbidity. Anger generated by drivers during driving can easily trigger road rage, which can seriously affect the life safety of drivers and other traffic participants, etc. In summary, emotions have a significant impact on all aspects of human life, so it is particularly important to identify them accurately.

At the present stage, the way of emotion recognition is mainly divided into two aspects; one is recognition through nonphysiological signals such as human facial expression,

voice tone, and body posture [2], because these non-physiological signals can be artificially controlled by means of camouflage and other means, resulting in sometimes not being able to obtain the real signal that can represent the emotion, thus not being able to accurately identify the real emotional state. On the other hand, physiological signals such as EEG signals, electro-ocular signals, ECG signals, EMG signals, and skin current responses can be used for emotion recognition [3, 4]. Emotion recognition based on physiological signals can obtain more objective and realistic results, which is also more conducive to practical applications [5].

An adverse state in the article is a combination of physical and mental reflections when people find that something real or imagined is beyond their expectations [6]. Chronic stress can lead to an increased susceptibility to illness, which can induce a variety of diseases [7]. In terms of emotions themselves, there is an important correlation between them and dysphoric states. Often, people wear black in bad states with a variety of complex emotions, of which

excitement, fear, and boredom are more common [8]. In contrast, emotions are expressed differently as a mental feeling and state, which often requires the use of language, tone of voice, facial expressions, behavioural gestures, breathing, and other media [9]. Physiological signals are more objective and realistic in reflecting people's emotional state and psychological feelings at the time [10].

The theory of "affective computing," which reflects specific emotions through changes in physiological signals, was first proposed by Professor Picard [11] at the MIT, who suggested the feasibility of extracting features from physiological signals for emotion recognition [12]. Nasoz et al. [13] from the University of Central Florida, USA, used  $k$ -nearest neighbour (kNN), Discriminant Function Analysis (DFA), and Marquardt back propagation (MBP). Kim et al. of Yonsei University, Korea [14], used the Support Vector Machine (SVM) algorithm; the Institute for Computational Science at the University of Augsburg, Germany, focused on comparing the recognition effects of combining different feature selection methods and classifiers [15]. The research on emotion computing in China started late, among which Guangyuan Liu's team from Southwest Jiaotong University conducted a comparative study on the effect of emotion recognition on emotion data samples from Augsburg University using a combination of various feature extraction and selection methods and classifiers [16].

Zhai et al. [17] used the SVM algorithm, and Setz et al. [18] carried out the classification using DFA and SVM algorithms. At present, there are relatively few studies at home and abroad on affective computing in adverse states, especially for different stress levels. The particle swarm optimisation (PSO) algorithm is combined with the kNN algorithm to investigate the identification of emotional experiences under adverse states based on multiple physiological signal parameters. Based on the removal of baseline emotions, the PSO algorithm optimises the selection of multiple features of multiple physiological signals and then uses kNN for classification to obtain better recognition results. The highest recognition rate reaches over 80%, which improves the correct rate of the traditional method of recognising emotional states with multiple physiological signals [19] and provides a way to explore the relationship between emotions in adverse states, and this provides a basis for exploring the relationship between emotions and multiple physiological signals in adverse states.

## 2. Research Methods for Identifying Emotions in the Adverse States

This study firstly designed different emotion and tension level evoking experiments under adverse states and collected six physiological signal parameters such as heart rate, respiration rate, skin impedance, blood oxygen saturation, pulse rate, and blood pressure under specific emotional states of multiple subjects in real time. Through pre-processing and feature extraction of these physiological data, combined with the results of the experimental subjective experience questionnaire, the PSO-kNN algorithm was used to select and classify the features of the experimental sample

data and finally to establish the emotion recognition model under the adverse state [20].

**2.1. Particle Swarm Algorithms.** The algorithm is conceptually simple, easy to implement, and fast to converge, has few parameter settings, and is little affected by changes in feature dimensions, making it an efficient search and optimisation algorithm [21]. Therefore, this paper uses the particle swarm algorithm for feature optimisation selection of physiological features.

Assuming that the total number of features is  $D$  and there are  $m$  individuals in the population, the velocity of the  $i$ th particle is  $V_i = (v_{i1}, v_{i2}, v_{i3}, \dots, v_{iD})^T$ , its position is  $X_i = (x_{i1}, x_{i2}, x_{i3}, \dots, x_{iD})^T$ , and the value of the position is a solution. By comparing the fitness values, the optimal position experienced by the current  $i$ th particle can be obtained as  $Pbest_i = (pbest_{i1}, pbest_{i2}, pbest_{i3}, \dots, pbest_{iD})^T$  and by comparing all particles, the optimal position of the whole population can be obtained as  $Gbest = (gbest_1, gbest_2, gbest_3, \dots, gbest_D)^T$ .

$$V_i^{n+1} = w \times V_i^n + C_1 \times \text{rand}_1() \times (Pbest_i - X_i^n) + C_2 \times \text{rand}_2() \times (Gbest - X_i^n), \quad (1)$$

$$X_i^{n+1} = X_i^n + V_i^n, \quad (2)$$

where  $w$  is the inertia weight factor, usually with values 0.4 to 0.9.  $C_1$  and  $C_2$  are the learning factors, and usually,  $C_1 = C_2 = 2$ .  $\text{rand}_1$  and  $\text{rand}_2$  are the random vectors between 0 and 1. A too-large inertia weight can increase the flight speed of the particles, which is conducive to jumping out of the local extremes, which makes the particles search locally. According to (3), let the inertia weight linearly decrease with the number for weight adjustment, faster to achieve convergence of the algorithm.

$$w(t) = w_{\max} - (w_{\max} - w_{\min}) \times \frac{T}{T_{\max}}, T = 1, 2, \dots, T_{\max}, \quad (3)$$

where  $w_{\max}$  is the maximum value of inertia weight,  $w_{\min}$  is the minimum value of inertia weight,  $T_{\max}$  is the maximum number of iterations, and  $T$  is the current number of iterations. The initial values of the parameters in this paper are set using the inertia weight method [22], where  $w$  will be initialized to a constant 0.729 and  $C_1 = C_2 = 1.494$ . To prevent particles from flying out of the search space,  $V_i \in [-V_{\max}, V_{\max}]$  is generally taken;  $V_{\max}$  will be too large to fly away from the best solution, and too small value will fall into a local optimum.

**2.2. The  $k$ -Nearest Neighbour Algorithm.** The kNN algorithm is a well-established and simple classification algorithm that makes full use of the physiological features of the entire emotion sample. The kNN algorithm, a commonly used classification algorithm, works on the principle that a sample is defined as belonging to a class if the vast majority of its sample points in the feature space belong to that class within a neighbourhood [23]. kNN algorithm is an algorithm in which the selected neighbours are all objects of the training

set that have been correctly classified. The nearest neighbour parameter is set to 1 in this paper.

**2.3. PSO-kNN Algorithm.** In the PSO-kNN algorithm [24], a particle is considered to have a higher fitness value when the number of features it produces is smaller and its classification accuracy is higher. The fitness function for evaluating each particle is  $f(x)$ . The larger the  $f(x)$  is, the better the fitness is, and the fitness function can be defined as follows:

$$\text{fitness} = \frac{1}{\text{RMSE} \times \text{Factor} + \text{Features}}, \quad (4)$$

where RMSE is the root mean square error, Features is the number of subsets of sample features, and Factor is the balance factor. We have the following steps:

*Step 1.* Design particles, represented by a binary bit string, with each binary bit corresponding to a feature in the physiological signal feature set, where a 1 in the bit indicates that the corresponding feature is in the selected feature subset and a 0 in the bit indicates that the corresponding feature is not in the selected feature subset [25].

*Step 2.* Initialize the particle swarm, i.e., set the  $X_i$  and initial velocity  $V_i$  of each particle at random.

*Step 3.* Learn and train with the kNN algorithm, and calculate the particle fitness according to equation (4).

*Step 4.* For each particle, compare the fitness function value  $f(x_i)$  with its own optimal value  $f(\text{pbest}_i)$ , and if  $f(x_i) < f(\text{pbest}_i)$ , replace the previous round's optimal value with the fitness value and replace the previous round's particle with the new one.

*Step 5.* Compare the best-fit value  $f(x_i)$  of each particle with the best-fit value  $f(\text{pbest}_i)$  of all particles. If  $f(x_i) < f(\text{pbest}_i)$ , replace the original global best-fit value with the best-fit value of that particle, while saving the particle.

*Step 6.* The particles according to model equations (1) and (2) of PSO produce a new population  $X_{i+1}$  with the following velocity adjustment rules: when  $v_i > V_{\max}$ ,  $v_i = V_{\max}$ ; when  $v_i \leq -V_{\max}$ ,  $v_i = -V_{\max}$ .

*Step 7.* Update the inertia factor  $\omega$ .

*Step 8.* Update the binary bits of the particle.

*Step 9.* Check the end condition. If it is satisfied, the search ends and the current optimal feature subset and classification accuracy are returned; otherwise, the number of iterations is increased so that iteration  $T = T + 1$  is reached and the search ends at the maximum number of iterations  $T_{\max}$  or the evaluation value is less than the given accuracy.

### 3. Emotion-Evoking Experiments

**3.1. Experimental Materials.** The International Affective Picture System (IAPS) [26] from the NIMH Emotion and Attention Research Center at the University of Florida was used as the main material for the different emotion

elicitation experiments in adverse states. These selected images were assessed for validity and arousal by a large number of subjects in different emotion elicitation experiments to determine the reliability of this approach. In this process, the validity and arousal are determined by the size of the defined data, where smaller numbers indicate lower validity and arousal and larger numbers indicate higher validity and arousal. Elicitation experiments for different levels of tension in the dysphoric state were elicited using different digit addition and subtraction mental arithmetic tasks. The efficacy values and arousal levels for the four IAPS emotionally arousing picture materials are given in Table 1.

A visual comparison of the effect of these four emotionally evocative picture material values and arousal levels is shown in Figure 1.

**3.2. Experimental Subjects.** There were 14 subjects (8 males and 6 females) from Shanghai Jiaotong University, aged 22 to 27 years old. They were physically and mentally healthy, had normal vision and hearing, had no previous history of psychiatric or neurological disorders, and had participated voluntarily in the experiment. They were not involved in strenuous exercise within 4 hours prior to the experiment and did not use any drugs within one week prior to the experiment. Before the start of the experiment [27], each subject was made fully aware of the purpose and procedure of the experiment and was tested with a stress-tolerance questionnaire, and all had a certain level of stress tolerance. The whole experiment was conducted in strict compliance with the Declaration of Helsinki.

**3.3. Experimental Equipment.** A high-performance computer system (Intel(R) Core™ i5-2310 CPU @2.90 GHz, 4 GB DDR3RAM, Lenovo, China; 17-inch professional display, 300 c d/m<sup>2</sup>, resolution 1280 × 768, vertical refresh rate 75 Hz) was used for the presentation of the emotionally evoked material. The screen for the presentation of pictures and mental arithmetic questions is approximately 50 cm away from the subject. Physiological signals are detected and recorded based on a portable multiphysiological parameter acquisition device developed by the laboratory, which can acquire a variety of physiological signal parameters such as ECG, heart rate, respiration rate, skin impedance, oxygen saturation, pulse rate, and blood pressure. The heart rate can be monitored from 30 bpm to 240 bpm with an error of  $\leq 2\%$ , respiratory rate  $\leq 5\%$ , skin impedance  $\leq 3\%$ , blood oxygen  $\leq 2\%$ , pulse rate  $\leq 3\%$ , and blood pressure within  $\pm 1.3$  kPa (10 mmHg).

#### 3.4. Experimental Procedure

**Experiment 1.** Firstly, in an emotion-evoking experiment with different visual stimuli, each of the ten emotion-evoking pictures of the same type was presented in sequence for 12 s. The whole process was completed in 2 min. The subjects took 2 min to calm down after each slide show and assessed the emotion elicited by the pictures. Before the start

TABLE 1: Efficacy values and arousal levels of the four IAPS emotion-evoking picture materials.

Emotions	Image RMS	Image wakefulness
Excitement	$6.0 \pm 0.5$	$6.5 \pm 0.5$
Boredom	$3.0 \pm 0.5$	$3.5 \pm 0.5$
Fear	$2.0 \pm 0.5$	$7.0 \pm 0.5$
Calm	$5.0 \pm 0.5$	$4.0 \pm 0.5$

of the experiment, each participant was given a set of pretests to familiarise them with the process and the experimental environment. The preexperiment images were also taken from the IAPS.

*Experiment 2.* For the elicitation of tension in different difficulty tasks, elicitation of tension in adverse states by giving two-digit, three-digit, and four-digit addition and subtraction mental arithmetic tasks with different levels of difficulty was done [28]. Each question was presented for 5 s, for a total of 125 s. Subjects were told to complete all questions as correctly as possible within the time limit, with an additional bonus if they obtained 95% or more score. Subjects were given a 2 min break between each set of questions to allow for emotional recovery. After the three sets of mental arithmetic tasks were completed [29], subjects were asked to give a subjective assessment of the level of tension induced by the three sets of tasks. Before the start of the experiment, a pretest was also conducted to familiarise the subjects with the procedure.

*3.5. Experimental Data Processing.* A total of 98 samples of physiological signals were obtained from 14 subjects through the tension elicitation experiment under different emotions and different difficulty tasks. Based on the subjects' subjective questionnaires [30], a total of 89 valid physiological signals were selected. Among them, 14 were calm, 10 were fearful, 12 were excited, 11 were bored, 14 were low tension, 14 were moderate tension, and 14 were high tension. In order to eliminate the differences in physiological data between subjects, the baseline physiological data of each subject in a calm emotional state were subtracted from the sample data obtained under fear, excitement, boredom, low tension, moderate tension, and high tension to obtain the baseline physiological sample data, i.e., 33 samples of the three types of emotions and 42 samples of the three tension levels. After completing the preprocessing of the data, feature extraction was performed on the various types of physiological signal data samples according to Table 2 and 33 features were finally obtained.

The specific distribution structure of the features extracted from the six physiological signals is illustrated in Figure 2.

#### 4. Experimental Results and Analysis

In the emotion elicitation experiments of the article, the emotion recognition algorithm performed on the basis of adverse states with multiple physiological signals was

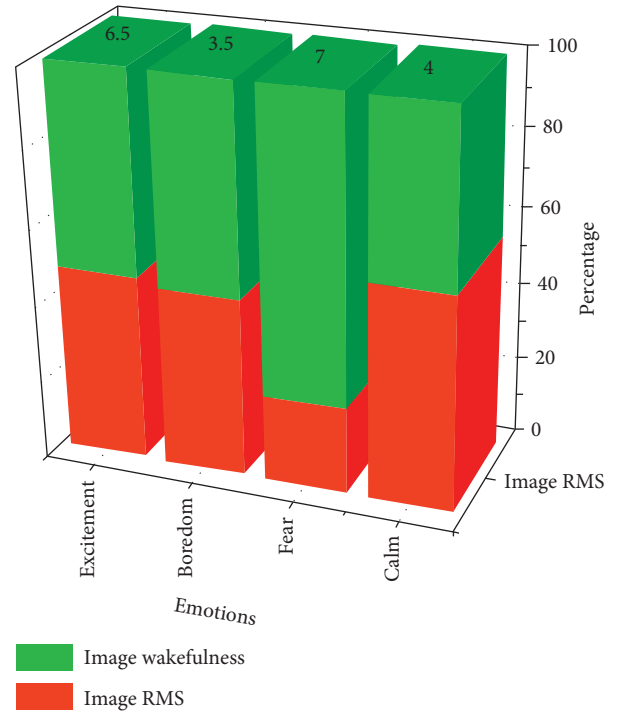


FIGURE 1: Visual comparison of the effect value and arousal of four different emotion-evoking picture materials.

TABLE 2: Characteristics of the six physiological signals extracted.

Physiological signals	Extraction characteristics
Blood pressure	Systolic, diastolic, and systolic-diastolic differential
Heart rate, respiratory rate, pulse rate, oxygen saturation, and skin impedance	Mean, variance, first-order difference mean, maximum, minimum, maximum-minimum difference

implemented on Matlab 2019a. For the 33 samples in the target, 21 of them were randomly selected as the training set and the remaining 12 samples were used to test the experimental results. In order to fully validate the experimental performance, we conducted the experiments separately for the samples with and without the removal of baseline data and the average results of the multiple experiments are shown in Table 3. ALL in the experimental data refers to the set of physiological signals containing BP, HR, RR, PR, SpO<sub>2</sub>, and SC.

The effect of the comparison of the average recognition rate of the un-baselined versus the baselined in the recognition results of the three types of emotional states and the three levels of tension in the adverse state is shown in Figure 3.

In order to identify the level of tension in poor states, 30 samples were drawn from a dataset of 42 tension level samples for training and the remaining 12 were used for testing. In order to conduct sufficient experiments to test the recognition effect in multiple situations, multiple

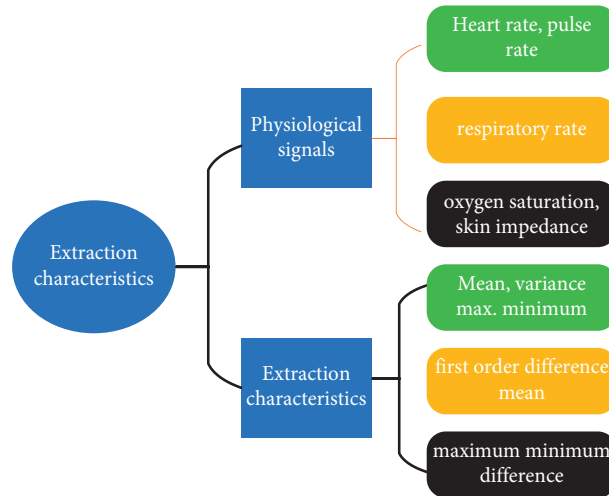


FIGURE 2: Structure of the specific distribution of features extracted from the six physiological signals.

experiments were conducted in both situations and the average of the multiple experiments was used as the final result. We conducted separate experiments on samples with and without the removal of baseline data, and the recognition results for the test set are shown in Table 4.

A comparison of the number of features of the optimal subset for the two cases is shown in Figure 4.

Particle swarm optimisation algorithms allow collaboration and information sharing between individuals in a population to find the optimal solution, which has the advantage of being simple and easy to implement and does not require many parameters to be adjusted. The results of kNN and PSO-kNN with the removal of baseline physiological data are presented in Table 5

A visual comparison of the recognition rates between kNN and PSO-kNN with the removal of baseline physiological data is shown in Figure 5.

The results of the kNN and PSO-kNN without the removal of baseline physiological data are shown in Table 6.

A visual comparison of the recognition rates between kNN and PSO-kNN without the removal of baseline physiological data is shown in Figure 6.

The IAPS picture system was used in the study here to design the emotion arousal experiment, which was evaluated using a subject emotion arousal questionnaire with high reliability. It can be concluded from the data in Tables 3 and 4 that when using the PSO-kNN algorithm for the identification of adverse emotions, the average recognition rates for all three different emotions were lower than the average

recognition rates for the three different levels of tension and that the data processing results without the removal of the baseline data were lower than the recognition results with the removal of the baseline physiological signal. This indicates that the removal of the baseline physiological signals can effectively improve the recognition of emotions in adverse states by eliminating the differences in physiological signals between individuals. Another very important finding is that the selected combination of signals is more accurate than the single signal feature.

In the training, diastolic blood pressure and heart rate maximum-minimum difference as well as pulse minimum were repeatedly selected as the optimal subset of features for emotion recognition. This indicates that the selected signals play an important role in the recognition of emotions in adverse states. Compared with the recognition results of Nasoz et al. using the kNN algorithm directly, this paper obtained better recognition results by combining the removal of baseline emotions with the PSO algorithm and optimising the selection of multiple features of multiple physiological signals before using kNN classification. Finally, based on the removal of baseline physiological signals, three predictions were made using the kNN-PSO algorithm for each of the three different emotions and three different levels of tension evoked by the experiment, and the predictions obtained are shown in Table 7.

Pie charts of the predicted outcomes for three of these different moods and three different levels of tension are shown in Figure 7.

TABLE 3: The classification of three stress emotions by physiological signals and their features' combination.

Physiological signals	Number of original features	Optimal subset feature number		Average recognition rate (%)	
		Not to baseline kNN	Removal of baseline kNN	Not to baseline kNN	Removal of baseline kNN
BP	3	2	1	41.67	50.00
HR	6	2	2	50.00	66.67
RR	6	3	3	58.33	66.67
PR	6	2	3	58.33	66.67
SpO <sub>2</sub>	6	1	1	33.33	33.33
SC	6	2	1	58.33	66.67
ALL	33	7	7	66.67	75.00

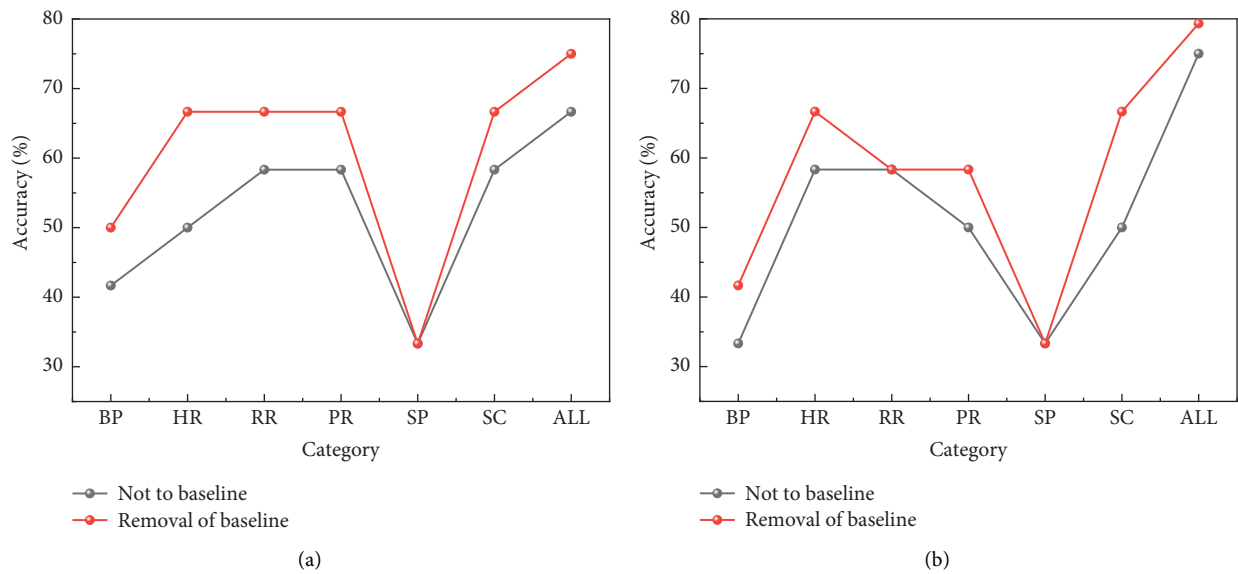


FIGURE 3: Comparison of the identification results of the (a) three types of emotional states and (b) three levels of tension.

TABLE 4: The classification of three tension degrees by physiological signals and their features' combination.

Physiological signals	Number of original features	Optimal subset feature number		Average recognition rate (%)	
		Not to baseline kNN	Go to baseline kNN	Not to baseline kNN	Go to baseline kNN
BP	3	1	2	33.33	41.67
HR	6	3	2	58.33	66.67
RR	6	2	1	58.33	58.33
PR	6	3	2	50.00	58.33
SpO <sub>2</sub>	6	1	1	33.33	33.33
SC	6	3	2	50.00	66.67
ALL	33	6	5	75.00	83.33

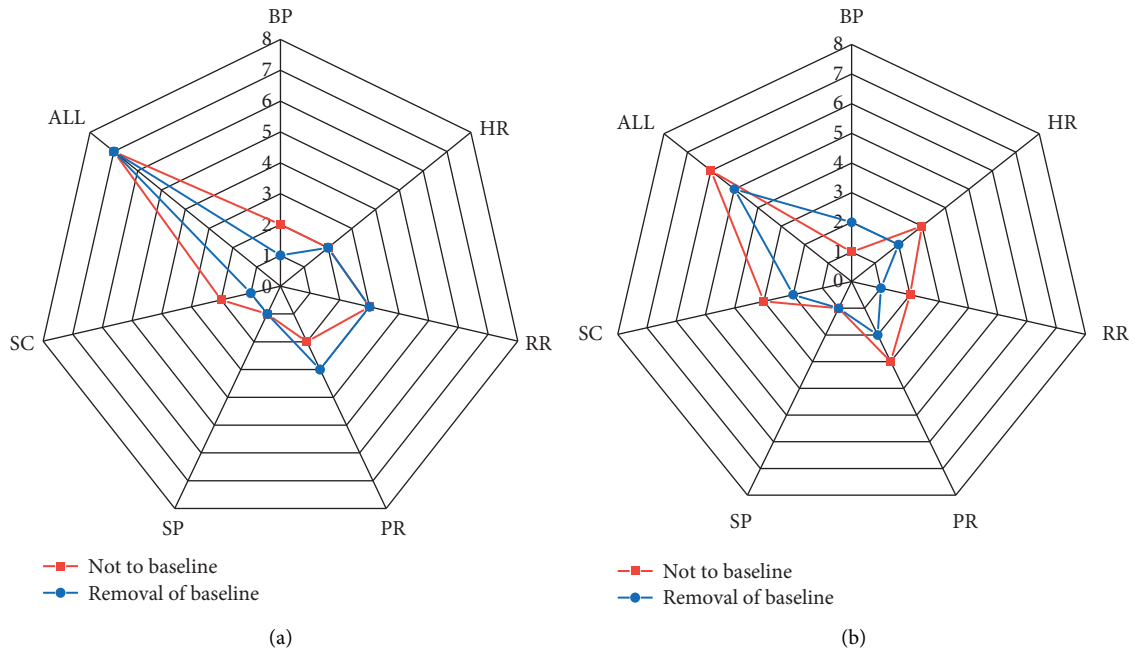


FIGURE 4: Comparison of the number of features of the optimal subset for the two types of cases: (a) three types of emotional states and (b) three levels of tension.

TABLE 5: Recognition rate results of kNN and PSO-kNN with the removal of baseline physiological data.

Category	Average recognition rate (%)	
	kNN	PSO-kNN
BP	50.00	66.67
HR	66.67	75.00
RR	66.67	66.67
PR	66.67	75.00
SpO <sub>2</sub>	33.33	50.00
SC	66.67	75.00
ALL	75.00	83.33

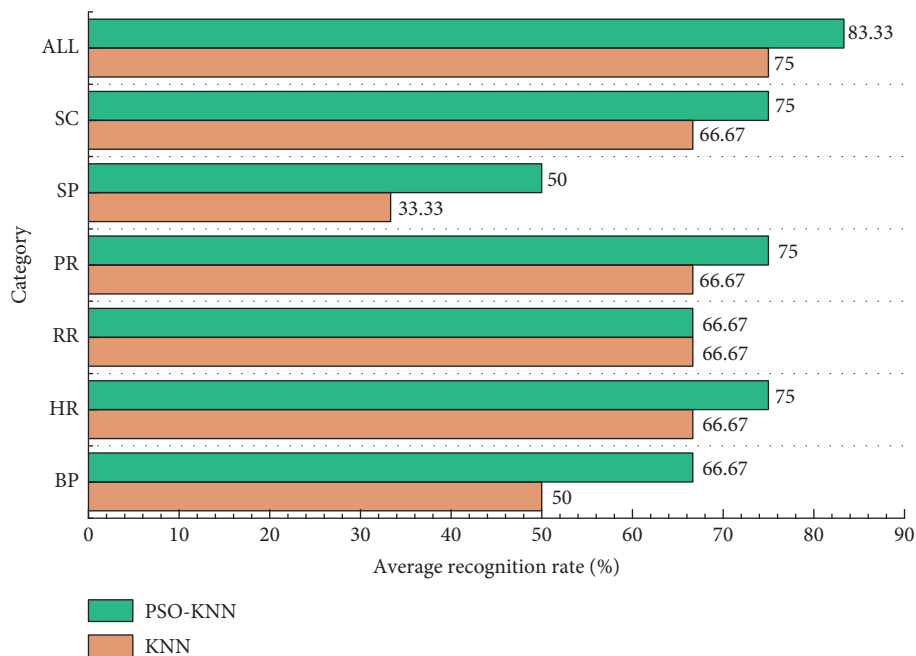


FIGURE 5: Visual comparison of recognition rates between kNN and PSO-kNN with the removal of baseline physiological data.

TABLE 6: Recognition rate results between kNN and PSO-kNN without removal of baseline physiological data.

Category	Average recognition rate (%)	
	kNN	PSO-kNN
BP	33.33	50.00
HR	58.33	75.00
RR	58.33	75.00
PR	50.00	66.67
SpO <sub>2</sub>	33.33	58.33
SC	50.00	75.00
ALL	75.00	83.33

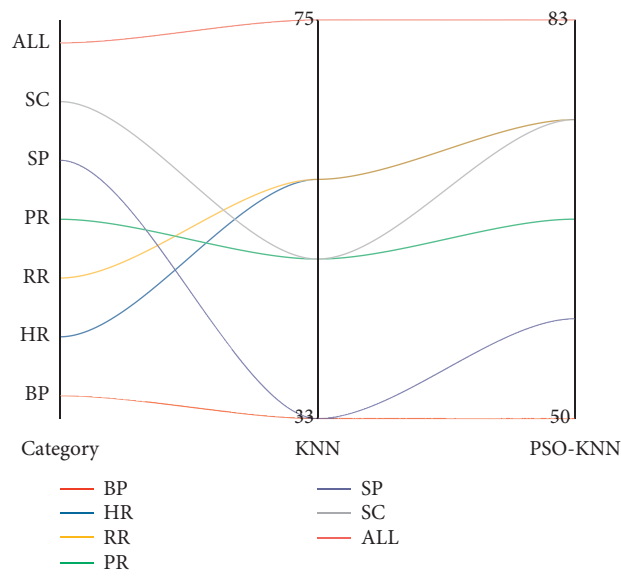


FIGURE 6: Visual comparison of kNN and PSO-kNN recognition rates without removal of baseline physiological data.

TABLE 7: Three sets of emotion-evoking experiments with three different emotions and three different levels of tension.

Experimental group	Accuracy (%)	
	Three stress emotions	Three tension degrees
1	75.00	83.33
2	83.33	91.67
3	83.33	91.67

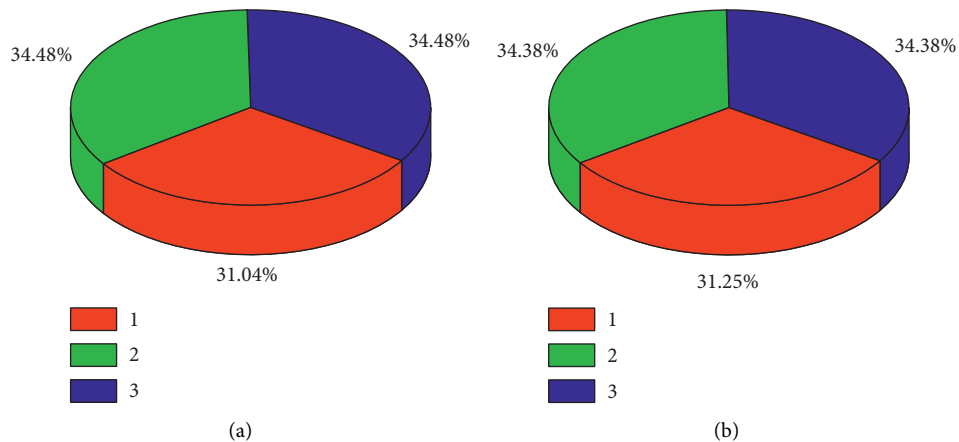


FIGURE 7: Pie charts of predicted outcomes for (a) three different emotions and (b) three different levels of tension.



## 5. Conclusions

In this paper, three types of emotions and three stress levels were induced using IAPS picture visual stimuli and mental calculation task experiments to build up a sample library of emotion-related physiological signals. Six feature vectors for effective recognition of stressful emotions were found through PSO feature optimisation, and the kNN algorithm was used to achieve emotion calculation and stress level recognition under adverse states. Experimental results show that the PSO-kNN algorithm achieves an effective recognition rate of 75% for the three emotions and 83.33% for the stress level. Through baseline data removal and PSO feature optimisation, the recognition results are better compared to the traditional kNN without feature optimisation selection. It provides some reference for the research of physiological signal processing and pattern recognition algorithms in affective computing research. Due to the limited sample data at present, it is difficult to test the deeper performance of the model. Future work will further expand the sample of emotion-related data in bad states, study the model with better performance for emotion recognition algorithm, and make the model work faster by considering the weighting relationship of the parameters in the PSO and kNN algorithms in more samples. Finally, we aim to extend the emotion recognition model to practical applications to make a real contribution to social change.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The author declares that there are no conflicts of interest.

## References

- [1] S. Wu, P. Anand, K. S. Cheung Simon, C. C. Ho, and S. Din, "Simulation of classroom student behavior recognition based on PSO-kNN algorithm and emotional image processing," *Journal of Intelligent and Fuzzy Systems*, vol. 40, no. 4, 2021.
- [2] P. Agethen, M. Otto, S. Mengel, and E. Rukzio, "Using marker-less motion capture systems for walk path Analysis in paced assembly flow lines," *Procedia Cirp*, vol. 54, pp. 152–157, 2016.
- [3] J. T. Spector and M. Lieblich, "Letter regarding. Comparison between low-cost marker-less and high-end marker-based motion capture systems for the computer-aided assessment of working ergonomics by Patrizi et al. and research reproducibility[J]," *Ergonomics*, vol. 60, no. 4, pp. 597–598, 2017.
- [4] K. Cahill-Rowley and J. Rose, "Temporal-spatial reach parameters derived from inertial sensors: comparison to 3D marker-based motion capture," *Journal of Biomechanics*, vol. 52, no. 52, pp. 11–16, 2017.
- [5] A. Ancillao, B. Savastano, M. Galli, and G. Albertini, "Three dimensional motion capture applied to violin playing: a study on feasibility and characterization of the motor strategy," *Computer Methods and Programs in Biomedicine*, vol. 149, no. 149, pp. 19–27, 2017.
- [6] V. Komisar, A. C. Novak, and B. Haycock, "A novel method for synchronizing motion capture with other data sources for millisecond-level precision," *Gait & Posture*, vol. 51, pp. 125–131, 2016.
- [7] F. S'a, A. Marques, N. B. F. Rocha, M. J. Trigueiro, C. Campos, and J. Schroder, "Kinematic parameters of throwing performance in patients with schizophrenia using a markerless motion capture system," *Somatosensory Research*, vol. 32, no. 2, pp. 77–86, 2015.
- [8] J. P. ngel-L'opez and N. A. de la Pe'na, "Kinematic HandAnalysis using motion capture technology," *IFMBE Proceedings*, vol. 28, pp. 257–260, 2015.
- [9] M.-K. Kim, T. Y. Kim, and J. Lyou, "Performance improvement of an AHRS for motion capture," *Journal of Institute of Control, Robotics and Systems*, vol. 21, no. 12, pp. 1167–1172, 2015.
- [10] Z.-M. Zhou and Z.-W. Chen, "A survey of motion capture data earning as high dimensional time series," *International Journal of Multimedia and Ubiquitous Engineering*, vol. 10, no. 9, pp. 17–30, 2015.
- [11] M. A. Khan, "Multiresolution coding of motion capture data for real-time multimedia applications[J]," *Multimedia Tools and Applications*, vol. 76, no. 15, pp. 1–16, 2016.
- [12] A. Puupponen, T. Wainio, B. Burger, and T. Jantunen, "Head movements in Finnish sign language on the basis of motion capture data," *Sign Language and Linguistics*, vol. 18, no. 1, pp. 41–89, 2015.
- [13] Y. Lee and H. Yoo, "Low-Cost 3D motion capture system using passive optical markers and monocular vision[J]," *Optik International Journal for Light & Electron Optics*, vol. 130, pp. 1397–1407, 2016.
- [14] M. M. Rahman, "Analysis of finger movements of a pianist using magnetic motion capture system with six dimensional position sensors," *Transactions of the Virtual Reality Society of Japan*, vol. 2, no. 15, pp. 243–250, 2017.
- [15] S. W. Park, H. S. Park, J. H. Kim, and H. Adeli, "3D displacement measurement model for health monitoring of structures using a motion capture system," *Measurement*, vol. 59, no. 59, pp. 352–362, 2015.
- [16] R. Giannetti, A. Petrella, J. Bach, A. K. Silverman, M. A. Saenz Nuno, and N. P. Mallada, "In vivo bone position measurement using high-frequency ultrasound validated with 3-D optical motion capture systems: a feasibility study," *Journal of Medical and Biological Engineering*, vol. 37, no. 7, pp. 1–8, 2017.
- [17] H. Zhang, L. Wang, and S. Chu, "Application of optical motion capture technology in power safety entitative simulation training system[J]," *Optics and Photonics Journal*, vol. 06, no. 8, pp. 155–163, 2016.
- [18] S. K. Wang, H. Xie, and B. Hu, "Research on protection property of running sportswear fabrics based on 3-D motion capture system," *Textiles and Light Industrial Science and Technology*, vol. 16, no. 3, pp. 57–62, 2014.
- [19] T. Miura, T. Kaiga, T. Shibata, and K. Tajima, "Low-dimensional feature vector extraction from motion capture data by phase plane Analysis[J]," *Journal of Information Processing*, vol. 25, no. 5, pp. 884–887, 2017.
- [20] D. M. Frost, T. A. C. Beach, T. L. Campbell, J. P. Callaghan, and S. M. McGill, "Can the Functional Movement Screen™ be used to capture changes in spine and knee motion control following 12 weeks of training," *Physical Therapy in Sport*, vol. 23, pp. 50–57, 2016.
- [21] A. Zhang, X. K. Yan, and A. G. Liu, "An introduction to A newly-developed "acupuncture needle manipulation training-

- evaluation system” based on optical motion capture technique,” *Acupuncture Research*, vol. 41, no. 6, pp. 556–559, 2016.
- [22] P. Tangsiridamrong and K. Mukdasai, “Stability and passivity Analysis of uncertain linear discrete-time systems with mixed interval time-varying delays,” *Dynamic Systems and Applications*, vol. 173, pp. 461–490, 2019.
- [23] Z. Jin, “The supply side reform mode of the integration development of industry and education of exhibition specialty in Chongqing Based on the Internet of things model,” *Dynamic Systems and Applications*, vol. 29, no. 3, pp. 413–425, 2020.
- [24] D. Holden, J. Saito, and T. Komura, “A deep learning framework for character motion synthesis and editing,” *ACM Transactions on Graphics*, vol. 35, no. 4, pp. 1–11, 2016.
- [25] E. C.-H. Lin, “A research on 3D motion database management and query system based on Kinect,” *Lecture Notes in Electrical Engineering*, vol. 329, pp. 29–35, 2015.
- [26] A. Paul, A. Ahmad, M. M. Rathore, and S. Jabbar, “Smart-buddy: defining human behaviors using big data analytics in social internet of things,” *IEEE Wireless Communications*, vol. 23, no. 5, pp. 68–74, 2016.
- [27] A. Paul, Y. C. Jiang, J. F. Wang, and J. F. Yang, “Parallel reconfigurable computing-based mapping algorithm for motion estimation in advanced video coding,” *ACM Transactions on Embedded Computing Systems*, vol. 11, no. S2, pp. 1–18, 2019.
- [28] M. M. Rathore, A. Paul, W.-H. Hong, H. Seo, I. Awan, and S. Saeed, “Exploiting IoT and big data analytics: defining smart digital city using real-time urban data,” *Sustainable Cities and Society*, vol. 40, no. 40, pp. 600–610, 2018.
- [29] A. Paul and R. Jeyaraj, “Internet of things: a primer,” *Human Behavior and Emerging Technologies*, vol. 1, no. 1, pp. 37–47, 2019.
- [30] A. Paul, A. Daniel, A. Ahmad, and S. Rho, “Cooperative cognitive intelligence for internet of vehicles,” *IEEE Systems Journal*, vol. 11, no. 3, pp. 1249–1258, 2017.

## Research Article

# Personalized Liver Cancer Risk Prediction Using Big Data Analytics Techniques with Image Processing Segmentation

Anurag Jain <sup>1</sup>, Ahmed Nadeem,<sup>2</sup> Huda Majdi Altoukhi <sup>3</sup>, Sajjad Shaukat Jamal,<sup>4</sup>  
Henry kwame Atiglah <sup>5</sup> and Haitham Elwahsh<sup>6</sup>

<sup>1</sup>Computer Science and Engineering Department, Radharaman Engineering College, Bhopal, Madhya Pradesh, India

<sup>2</sup>Department of Pharmacology & Toxicology, College of Pharmacy, King Saud University, PO Box 2455, Riyadh 11451, Saudi Arabia

<sup>3</sup>Affiliation: Department of Radiology, Faculty of Medicine, King Abdulaziz University Hospital, Jeddah, 21589, Saudi Arabia

<sup>4</sup>Department of Mathematics, College of Science, King Khalid University, Abha, Saudi Arabia

<sup>5</sup>Department of Electrical and Electronics Engineering, Tamale Technical University, Tamale, Ghana

<sup>6</sup>Computer Science Department, Faculty of Computers and Information, Kafrelsheikh University, Kafrelsheikh, Egypt

Correspondence should be addressed to Henry kwame Atiglah; [hkatiglah@tatu.edu.gh](mailto:hkatiglah@tatu.edu.gh)

Received 27 November 2021; Revised 29 December 2021; Accepted 29 January 2022; Published 28 March 2022

Academic Editor: Ahmed A. Ewees

Copyright © 2022 Anurag Jain et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A technology known as data analytics is a massively parallel processing approach that may be used to forecast a wide range of illnesses. Many scientific research methodologies have the problem of requiring a significant amount of time and processing effort, which has a negative impact on the overall performance of the system. Virtual screening (VS) is a drug discovery approach that makes use of big data techniques and is based on the concept of virtual screening. This approach is utilised for the development of novel drugs, and it is a time-consuming procedure that includes the docking of ligands in several databases in order to build the protein receptor. The proposed work is divided into two modules: image processing-based cancer segmentation and analysis using extracted features using big data analytics, and cancer segmentation and analysis using extracted features using image processing. This statistical approach is critical in the development of new drugs for the treatment of liver cancer. Machine learning methods were utilised in the prediction of liver cancer, including the MapReduce and Mahout algorithms, which were used to prefilter the set of ligand filaments before they were used in the prediction of liver cancer. This work proposes the SMRF algorithm, an improved scalable random forest algorithm built on the MapReduce foundation. Using a computer cluster or cloud computing environment, this new method categorises massive datasets. With SMRF, small amounts of data are processed and optimised over a large number of computers, allowing for the highest possible throughput. When compared to the standard random forest method, the testing findings reveal that the SMRF algorithm exhibits the same level of accuracy deterioration but exhibits superior overall performance. The accuracy range of 80 percent using the performance metrics analysis is included in the actual formulation of the medicine that is utilised for liver cancer prediction in this study.

## 1. Introduction

The liver is the second-largest organ in the human body after the skin. Approximately three pounds is the weight of a healthy adult's liver. The liver is situated on the right side of the body, under the right lung, and is covered by the ribcage [1]. A sulcus separates each of the lobes (a ridge). This situation is similar to that of a chemical factory. The liver's role in digestion is to produce proteins and bile, both of which the body needs to function effectively, the removal of

toxins from the body that have been eaten [2]. By using vitamins, carbohydrates, and minerals stored in the liver, it is able to break down numerous nutrients from the gut while also controlling cholesterol excretion. It also produces rapid energy when needed. Throughout the body, the cell serves as the fundamental unit that constructs the tissues. Growing and dividing into new cells are typical functions of cells in their normal state [3]. The cell is replaced with a fresh one if it gets old or broken. Every now and again, something goes wrong during the operation. In contrast to the fact that the

body does not manufacture new cells, nodules and tumours are produced by the tissues of old or damaged cells. Liver tumours are classified into two types: benign and malignant [4]. In comparison with malignant tumour, benign tumour is less dangerous. Tumours that are not damaging to the patient's life are benign tumours, which are very uncommon. They are not usually re-grown after it has been excised, unlike malignant tumours. However, it does not spread to other parts of the body; instead, it attacks tissues in their immediate surroundings. Tumours that are malignant are malignant tumours, which are cancerous and may be fatal [5]. When it is removed from the body, it re-grows and becomes very dangerous. A stomach or intestinal infection may be lethal and spreads throughout the body, affecting many organs. Primary liver cancer and secondary liver cancer are the two forms of liver cancer that may occur in people. Primarily, liver cancer refers to a tumour (malignant) that begins in the liver itself. It is probable that secondary liver cancer develops in another place of the body and then spreads into the liver [6]. Hepatocellular carcinoma (HCC) is the term used to describe a tumour that develops in hepatocyte cells. Cancer of the liver that has developed from inside the organ itself. Hepatocellular carcinoma is responsible for around 75–90 percent of all liver cancer cases in the United States. Primarily, liver tumours are classified into several categories, including cholangiocarcinoma or two-bile-duct cancer, coupled HCC and cholangiocarcinoma tumour of mesenchymal tissue, sarcoma, and hepatoblastoma. In children and young adults [7], this uncommon malignant tumour manifests itself.

Based on the insights achieved, new technologies in the computer science sector are expected to emerge in the next years. The “third paradigm” is derived from the many analyses and implementations that have been carried out [8]. The findings in biomedical applications were obtained due to the experimental analysis and the numerous surveys that were carried out during the research process. Various discoveries have been developed to fulfil the needs of the imaginative future and keep up with the ever-increasing number of requirements. The data processing complexity increases as a result of the speed parameter being used [9]. In this case, the study is concentrated on developing applications that benefit from an increase in the speed of computation and an increase in available computing resources. Gathering and processing of a wide range of data is the primary reason for the development of this paradigm, which is beneficial to researchers [10]. In the fields of medicinal applications and biomedical research, some of the most significant breakthroughs have been made. The development of new drugs is a complicated process that involves a variety of procedures. Various molecular structures were chosen and identified from among the  $n$  number of potential possibilities (see Figure 1). The time consumption in the discovery of biological applications, which was endured for 10 to 15 years after the discovery [11], was documented. As the number of ligands available in the pharmaceutical industry grows, a big data analytics technology called virtual screening (VS) [12] is being used to screen them all. The primary goal of the approach that has been established is the

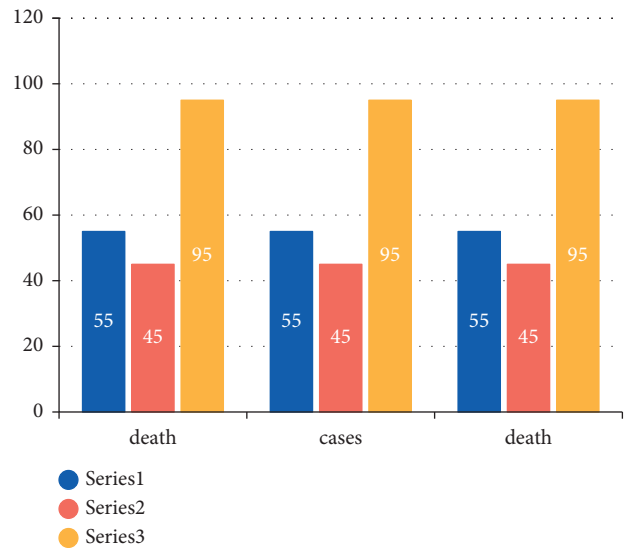


FIGURE 1: Number of deaths due to liver cancer in developed and developing countries.

prediction of ligands in order to find the protein receptor. Using the docking technique, it is possible to shorten the amount of time it takes to identify new medications for the treatment of liver cancer. Hepatocellular carcinoma (HCC) is the most difficult kind of cancer to treat since it develops in the liver's tissue and is very harmful in today's society. Global liver cancer is one of the types of liver cancer that has increased from 641000 to 643000 in the last four decades [13]. Figure 1 depicts the mortality rates and the increment in liver cancer in developing nations and developed countries, respectively.

Extensive data analysis can help speed up the process of medication development, which is a time-consuming endeavour. To provide an example, the creation of aspirin, which is used in biomedical therapy, was inspired by a study of patients' electronic health records (EHRs) who were contaminated [14]. In this study, the records of the patients were gathered from the database of the United States Preventive Services Task Force, which use aspirin to treat cancer cells. In addition, raloxifene [15], which was approved by the FDA in 2007, and dapoxetine, used for the diagnosis of ejaculation are examples of medications that have been approved. Using healthcare informatics software, a large portion of the therapeutical sector has examined gene expression and cellular screening in order to determine the chemical makeup of the cancer cell [16]. As an update, numerous conversations have taken place in the biomedical sector in preparation for the drug development process that will be discussed in the following sections.

Speed-up learning is a sort of machine learning in which the problem solvers solve the issue based on their previous expertise [17]. It examines the previous problem solver's experience and traces their steps and solutions. A distinction is made between rote and explanation-based learning. Roughly speaking, rote learning is the more traditional approach, finding out via getting advice. In this sort of learning, the advice may come from a variety of sources,

such as human experts and other internet-based information [18]. Learning by example is an inductive learning method in which the decision tree is utilised to guide the learner through the process. This algorithm is based on Quinlan's algorithm, which is also known as ID3. It is the process of inductive learning in which the unlabelled data are grouped in comparable groups called clusters using the Euclidean distance and the Manhattan distance as a basis for grouping [19]. Similarly, to inductive learning, learning by analogy is a kind of learning in which information is retrieved from previous knowledge. It is one of the most basic deduction strategies in human cognition. The rest of the article is organized as follows: Section 2 represents the background analysis, Section 3 represents the proposed work, and Section 4 represents the experimental study, and Section 5 represents the conclusion and future work.

## 2. Background Analysis

It was necessary to add several software and technologies in order to create the new medication. In the suggested portion, numerous platforms from the current structure were explored in detail, allowing for a more in-depth examination of the planned work.

*2.1. Hadoop/MapReduce Technique.* Through the utilisation of enormous datasets, the MapReduce approach [20], an advanced and rarely used technique in the IT sector, is employed in big data analytics. The MapReduce technique is an advanced and seldom used technique in the IT field. A large number of nodes can benefit from parallel and distributed MapReduce execution because of the technique's high scalability and reliability [21]. MapReduce is a method that is straightforward in terms of programming, and it is widely utilised in a variety of real-time applications. The MapReduce approach used to handle a large amount of data at one time. The key benefit of the MapReduce approach is that it is easy to install and has a lower level of fault tolerance than other techniques. The most important job is to establish a model for the discovery of a new medication [17]. The MapReduce approach, which is used to identify new drugs, makes use of two processes, namely, the map function and the reduction function.

*2.2. Mahout Technique.* Apache Mahout, a key approach discovered by the Apache Foundation that leverages the library function of machine learning algorithms in conjunction with the Hadoop platform as its foundation, is a major technology. Mahout has been at the forefront of new and innovative developments since the various algorithms were implemented [22]. Mahout is used for big data processing data structures that are compatible with a single machine learning approach, such as deep learning. Despite the fact that this methodology includes the Java library function, it does not include the user interface structure [23].

*2.3. Open Babel Technique.* In order to examine the varied chemical compositions of the obtained data, the chemical

expert created Open Babel, which is an open-source programme that is available for free. The primary goal of this programme is to construct multiplatform libraries for molecular models [24], as well as to do various data conversions for the medicine that has been produced.

The research [17] indicated that back propagation produced the greatest results in terms of accuracy (71.59 percent), precision (69.74 percent), and specificity (82 percent). The NBC classifier has much better sensitivity (77.95 percent) than the other classifiers. The KNN technique, when applied to the AP Liver dataset and using common characteristics (SGOT, SGPT, and ALP), provides a high accuracy when compared to other algorithms. ANN and SVM performance were evaluated on various cancer datasets in this study [18], with accuracy, sensitivity, specificity, and area under the curve all being measured and compared (AUC). The BUPA liver disorder training set (70 percent) and testing set (30 percent) were chosen, and after analysis, SVM provided (accuracy, 63.11 percent; sensitivity, 36.67 percent; specificity, 100.0 percent; AUC, 68.34 percent) and artificial neural networks provided (accuracy, 63.11 percent; sensitivity, 36.67 percent; specificity 100.0 percent, and AUC 68.34 percent) (accuracy, 57.28 percent; sensitivity, 75.00 percent; specificity, 32.56 percent; AUC, 53.78 percent). In research [19], a dataset of 78 percent of liver cancer patients associated with cirrhosis was employed, which included two forms of liver cancer: HCC and nontumour livers. The data were separated into two groups: training and testing. The K-nearest neighbour approach was used to eliminate the values that were missing. Employing principal component analysis, the author optimised a fuzzy neural network before comparing the GA search results to the improved fuzzy neural network. In this study, it was discovered that using a smaller number of genes, FNN-PCA could achieve an accuracy of 95.8 percent. The classification of the liver and nonliver disease datasets was based on the findings of this study [20]. Medical data from a Chennai medical centre with 15 features were used for preprocessing, and the C4.5 and Naive Bayes classifiers were used for the study. The C4.5 algorithm outperformed the Naive Bayes method in terms of accuracy.

The major contributions of the proposed work are to identify cancer using both features obtained using map reducing technique and image processing is used to identify the classes of cancer in the patients' CT scans, and to reduce execution time and enhance the accuracy rate.

## 3. Proposed Approach

In the following section, the technique has been developed to discover the new drug for the treatment of liver cancer in the field of big data analytics [25]. The approach was made in the initial stage of dataset selection and the algorithm discovery in the big data society.

*3.1. Dataset.* The liver cancer diagnosis is based on the protein deficiency. The protein deficiency of the liver tissue is identified using the 4JLU receptor, which includes the crystal

structure of BRCA1 [12]. The protein data bank is used to obtain the structural information needed to conduct the study (PDB). Because the receptor had to be built from the ground up, the Cambridge library's collection of ligands was used [13]. This different ligand contains  $10^6$  ligands, of which the proposed work randomly collects  $10^4$  ligands. Virtual screening process is carried out with the use of AutoDock Vina (AV). Input images are acquired from the Kaggle dataset to extract the features of cancer from the tumour sets.

**3.2. Virtual Screening (VS).** The liver cancer features are collected from the infected and dis-infected data. Figure 2 represents the frequency of the affinity (kcal/mol) and shows the median as the separation point.

The separation point is taken using the median point which is calculated as  $-5.8$  Kcal/mol. The difference between the active and inactive ligands is used to compute the separation point. Using the real positive and true-negative values, the greater value is arrived at [26]. The false number is neglected using the mean value. The molecular format PDBQT [27] is the complex structure, which converts the ligands to the fingerprint format (FPF). This converts the various algorithms into machine learning algorithm. Open Label is the toolbox, which is considered to follow the chemical composition of the discovered drug. This will follow the chemical conversion taking place in the drug structure. The FPF, which is hexadecimal in structure, is converted into a binary structure that comprises  $n * m$  matrix formats. The vector element is created and transferred towards the label class of the dataset [28].

From the pseudocode, the first stage, like with many other ground filtering methods, is the production of  $V_{i_{min}}$ , which is based on the cell size parameter and the amount of data present. It is possible to provide the two vectors corresponding to  $[min: cellSize: max]$  for each coordinate— $x_i$  and  $y_i$ —directly from the user's input or to quickly and automatically compute them from the data provided. Instead of generating a raster for each of the  $(x, y)$  dimensions, the SMRF method generates a raster spanning the ranges between the ceiling and floor of the lowest and maximum values for each dimension. If the cell size parameter is not an integer and is not specified, the same general rule applies to values that are evenly divided by the cell size parameter. Using the previous example, if the cell size is equal to 0.5 m and the  $x$  values are in the range 52345.6 to 52545.4, the range would be [52346 52545].

It is designed to be applied to both the first and final returns of the point cloud, while it is possible to build a minimal surface that is almost as good with just the latest returns, as stated in the next paragraph. However, even though the last return of any given pulse is most likely to be ground, this is not always the case: for example, it is possible that the last return of one pulse happens to hit an object at a given location, while the first return of another pulse happens to strike closer to the ground at the same location. A minor inaccuracy would be introduced into the DEM as a result of the early removal of the first return from the second pulse in this example, which would be impossible to remove

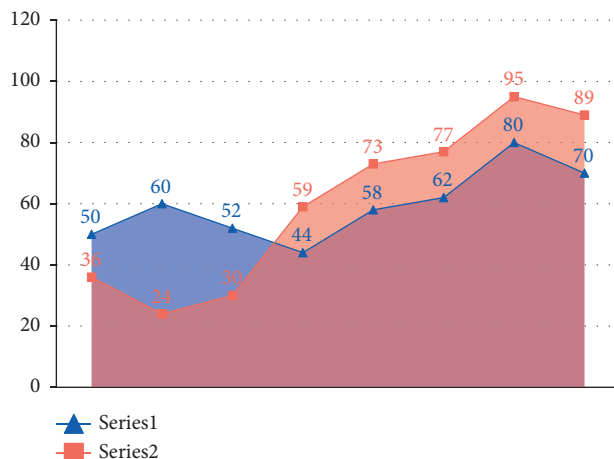


FIGURE 2: Variation in the frequency against the binding affinity interval in Kcal/mol.

with any filter. Therefore, it is recommended that both the first and last returns be utilised, since the unnecessary observations are quickly deleted during the first grid-generation process.

The minimal surface grid  $V_{i_{min}}$  created by the vectors  $(x_i, y_i)$  is filled with the elevation data that are closest to the original LIDAR data and is the lowest elevation.

**3.3. Proposed Model Architecture.** The data construction step is followed by the five-model formation. This model is used to train the dataset with the labelled class that is used to predict the severity of the cancer using machine learning algorithm [14]. The prediction is made for the discovery of a new drug with certain chemical composition.

Figure 3 represents the flowchart of the proposed work. The implemented algorithm is based on the MapReduce algorithm using the Java implementation. In the proposed work, the best three algorithms were selected and combined to form the classifier with the higher accuracy.

**3.3.1. SMRF (Scalable MapReduce Random Forest).** The electronic health records include information such as the patient's identification number, status, age, gender, hepatitis, ascites, edema, billi, cholesterol, albumin, and other vital signs. The data under consideration must be clinically converted, that is, made acceptable for further processing, before it can be used. The clinical transformation stage is also referred to as the preparation step in certain circles.

Null values, irrelevant values, and noisy values may be found in the unprocessed data. These data flaws would result in misclassification, and as a result, they would need to be converted therapeutically. Mode function is used to impute missing data from the considered dataset with values generated using the mode function.

Following the preparation of data, three subsets from the datasets are prepared for use in the random forest classification system for categorising occurrences.

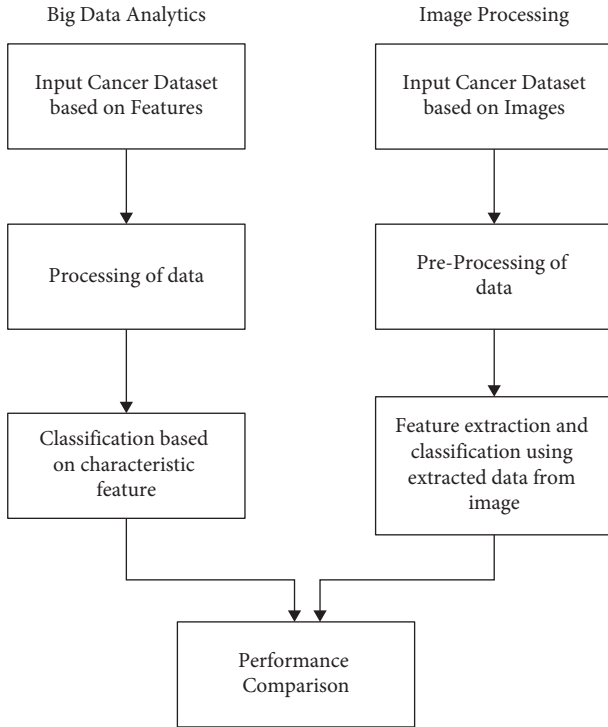


FIGURE 3: Flowchart of the proposed work.

When generating the subgroup, three characteristics will be taken into consideration: platelet count, alkaline phosphate, and cholesterol levels.

The random forests are constructed by combining three classification techniques, namely, C4.5, J48, and Naive Bayes, into a single structure. There are many other voting methods that may be used for an ensemble of classifiers; however, in this case, we will use the majority vote technique to execute voting with a variety of classifiers. The ultimate conclusion of the majority of classifiers will be shown as the output in this case.

Random forest is one of the machine learning techniques that is constructed using the multilayer of decision trees. This method is developed using the bagging process [29]. The independent variable  $X$  is considered, which is combined with the decision tree  $K$  to form the classification matrix of  $h_1(X), h_2(X), \dots, h_k(X)$ . Each of the classifier is trained and classified using the matrix obtained in the classification process. SMRF (scalable MapReduce random forest) is one of the techniques of the big data learning [15]. This proposed technique consists of three phases, which is implemented as follow:

*Step 1.* The descriptor file from the dataset is subjected towards the attribute description.

*Step 2.* It is represented as the generating stage and subdivides the given dataset into bootstrap samples that can be trained using the bagging algorithm

*Step 3.* It is represented as the voting phase where the decision trees give the classification results. The proposed

SMRF technique decides the decision of the classification with the higher voting technique. Figure 4 shows the scalable random forest algorithm based on MapReduce technique.

Bayes theorem—It is of importance to determine which theory is the most likely for given space  $S$ . In the context of machine learning, the term is defined by the observed training data.  $P()$  is the initial probability that the hypothesis is true before any training facts are learned, and  $P()$  is the prior probability that the hypothesis is true before any background knowledge is learned about the right hypothesis. Presumptions may have some prior knowledge depending on the facts given, even if no prior information is available. In a similar vein, prior probability ( $\alpha$ ) on the provided training data is calculated.  $q(\alpha)$  will represent the probability based on the supplied data. In general, the probability of  $x$  provided by  $y$  may be represented as  $Q(x|y)$ , which stands for probability of  $x$  given by  $y$ . If you are interested in machine learning, the portion of interest is  $Q(\beta)$ , which is the posterior probability on a hypothesis based on a particular training dataset, which may be used to determine the confidence in a given dataset [16]. The base theorem is the cornerstone of the Bayesian learning approach because it calculates the posterior probability  $Q(\beta)$  from the prior probability,  $Q(\alpha)$  and  $Q(\beta)$  being the probabilities of the past and future. The Bayes theorem is a mathematical formula that predicts the likelihood of an event:

$$Q(\alpha|\beta) = \frac{Q(\beta|\alpha)Q(\alpha)}{Q(\beta)}. \quad (1)$$

According to Bayes' theorem,  $Q(|)$  grows as  $Q(|)$  and  $Q(|)$  increase in importance. If  $Q(|)$  grows, it can be observed from the equation that the value of  $Q(|)$  decreases. Most likely, the observed will be independent of the observed. The  $S$ -hypothesis will be the most likely one to be tested based on the observed facts. When the most likely values are selected, the hypothesis known as the Maximum A Posteriori Bayesian Inference Data Prior Information Statistical Conclusion (MAP) hypothesis is used. When computing each candidate hypothesis, this approach makes use of the Bayes theorem:

$$\begin{aligned} & \operatorname{argmax}_{\alpha \in S} Q(\alpha/\beta), \\ & \operatorname{argmax}_{\alpha \in S} \frac{qQ(\beta/\alpha)q(\alpha)}{q(\beta)}, \\ & \operatorname{argmax}_{\alpha \in S} q(\beta/\alpha)q(\alpha). \end{aligned} \quad (2)$$

In the final step,  $q(\beta)$  is removed since it is not reliant in any way and acts as a constant.

*3.4. K-Means Clustering-Based Segmentation.* Making use of training cases and test instances to choose functions that are comparable, the distance function used by  $K$  star is based on entropy, which is distinct from other distance functions. Instance-based learning categorises instances from a database of previously categorised examples. It is anticipated that

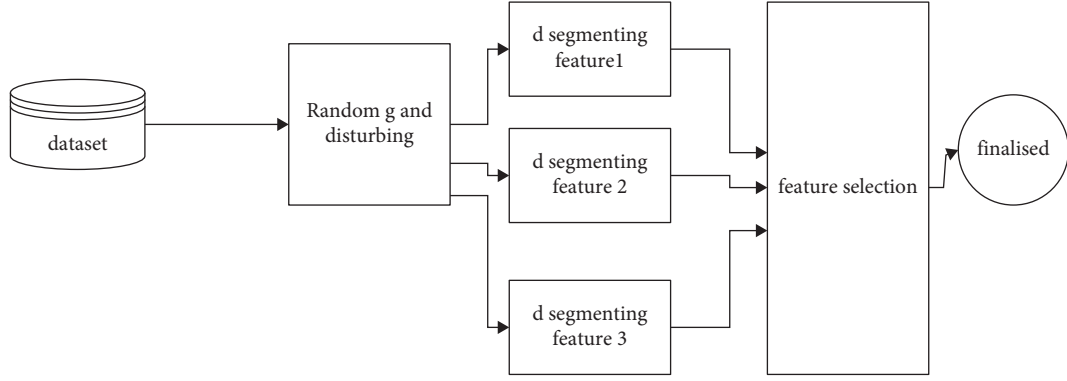


FIGURE 4: Scalable random forest algorithm based on MapReduce.

occurrences that are comparable to one another would have the same categorisation as one another. K star work utilises transformation, which picks one instance of a transformation at random from all of the possible transformations using the entropic measure. Entropy is employed as a distance metre in this approach, and the distance between the instances is computed using it. The complexity of a transformation is measured by the distance between occurrences of the transformation. It was accomplished via the use of instance transforms and mappings for a limited number of transformations. Assume that is the initial position and that is the ending point. Let us suppose that  $X$  is predefined and that there are an infinite number of points. Let  $x$  equal  $X$ ; then,  $x$  will be the map  $x : y$ . The map instance itself is denoted by the symbol  $X(=)$ , and  $q$  is terminated.  $q$  is a transformation on, and it has a single definition. Explanation:  $x(n) = xn$  ( $1 \dots x1$  ( $n \dots$ )), where  $x$  is the number of elements in the set. Then,  $x1 \dots xn$  is the number of times  $x$  equals  $x1 \dots xn$ . When  $q$  is a probability function  $X^*$ , it means that it should satisfy the requirements of the following qualities:

$$\begin{aligned}
 0 &\leq \frac{q(\bar{x}v)}{q(\bar{x})} \leq 1, \\
 \sum_v q(\bar{x}v) &= q(\bar{x}), \\
 q(\Lambda) &= 1, \\
 \sum_{x \in q} q(\bar{x}) &= 1 \dots
 \end{aligned} \tag{3}$$

$r^*$  is the probability function that defines all paths moving from  $\alpha$  to  $\beta$ . As mentioned the probability function  $q^*$  which is defined as the probability of all tracks from instance  $a$  to instance  $b$ :

$$r^* \left( \frac{\beta}{\alpha} \right) = \sum_{x \in r: x(\alpha)=\beta} r(x). \tag{4}$$

$r^*$  satisfies following properties:

$$\begin{aligned}
 \sum_{\beta} r^* \left( \frac{\beta}{\alpha} \right) &= 1, \\
 0 &\leq r^* \left( \frac{\beta}{\alpha} \right) \leq 1,
 \end{aligned} \tag{5}$$

The  $L^*$  function is then defined as

$$L^* \left( \frac{\beta}{\alpha} \right) = -\log_2 r^* \left( \frac{\beta}{\alpha} \right). \tag{6}$$

#### 4. Experimental Results

The proposed SMRF technique is performed using the Hadoop environmental factors. The Java workbench is adopted to run the random forest algorithm with the same parameters of the traditional algorithm. The system's precision is determined by the parameters marked as  $K$ . To compare the various algorithms with the proposed approach, many methodologies were investigated. The mean value of the proposed work determines the accuracy of the system. The experimental analysis of various applications was considered to analyse the proposed work that is tabulated in Table 1.

The experimental analysis of various applications is shown in Table 1. In the various analyses, the proposed SMRF algorithm has the better accuracy in various fields and lesser error factor. Figure 5 represents the comparison of the proposed algorithm with the traditional algorithm.

For SMRF, the accuracies in datasets "corral" and "ionosphere" are 97.66% and 93.16%, respectively, which are much higher than the traditional random forest. The experimental results with the mean parameter, that is represented as  $K$ , are shown in Figure 5. The proposed algorithm has 10 nodal points with the 100-decision tree structure. The SMRF algorithm has parallel performance, which reduces the classification timings and increases the system's accuracy based on the MapReduce model. Scalability of the system is higher when compared to the other algorithms. This proposed work results in the good accuracy in the classification that would yield the better drug discovery.

**4.1. Image Acquisition.** Database images are collected from the cancer imaging Archive, which consists of both normal and abnormal images. The database images consist of MRI images and CT scan images, as well as ultrasound scan images. These images are the collection of both normal lung and abnormal lung. The proposed work consists of around



TABLE 1: Classification-based data analytics.

Datasets	SMRF (%)	Traditional RF (%)
Liver	95.23	96.55
Cancer	94.35	92.83
DNA	99.16	99.53
Chess	97.66	81.25
Corral	93.16	88.03
Ionosphere	92.00	92.00
Iris	95.00	87.70
Letter	90.80	85.35
Satimage	95.84	93.50
Segment	99.98	99.95
Shuttle	95.67	93.32

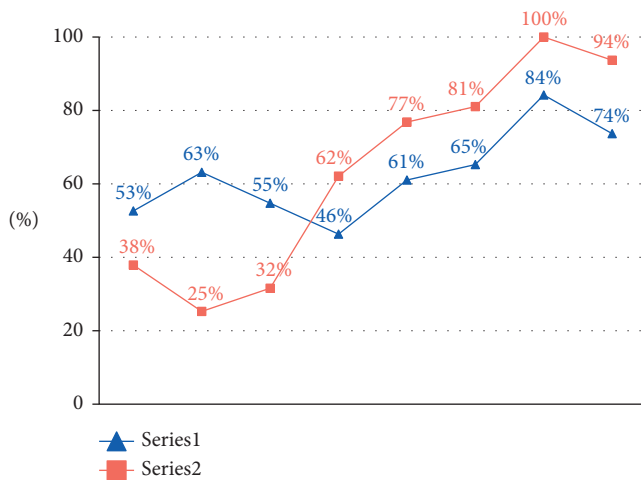


FIGURE 5: Comparison of the proposed SMRF with various applications.

300 images, which include MRI, CT scan, and ultrasound images. The input images are shown in Figure 6.

**4.2. Morphological Operations.** Morphological operations consist of categories such as close, erosion, dilation, mask, and mark. These procedures are carried out to smoothen the dilated area and to remove the unwanted particles within the converted RGB image. Using these techniques, the filtered picture may be separated into its parts by structural and morphological procedures. The output results of this process in MRI scan, CT scan, and ultrasound scan is shown in Figure 6.

Figure 7 represents the preprocessing stage in cancer images.

**4.3. Segmentation.** The segmentation process is based on the watershed algorithm and Sobel edge detection technique. The watershed algorithm is a mathematical morphology method founded on topology conception and may just belong to the region-founded segmentation approaches. Its intuitive proposal originates from the topography; photos are viewed as a topology remedy within the topography; and the grayscale value of each pixel on images stands for the elevation at this point. For the watershed algorithm, there

are numerous calculation approaches; an effective algorithm [7] based on immersion simulation proposed by Vincent and Soille is a milestone of the watershed algorithm study, for it improves an order of magnitude in calculation when put next with the long-established watershed algorithms, and for this reason, the watershed algorithm has been applied largely. Thus, the results of watershed segmentation are shown in Figure 8.

**4.4. Classification.** Consider the following scenario: the input picture is of an elephant. This picture, complete with pixels, is the first image to be put into the convolutional layer system. A black-and-white image is read as a 2D layer, with each pixel given a value between zero and two hundred and fifty-five (255), with zero being entirely black and two hundred and fifty-five representing fully white. For a colour image, on the other hand, the result is a 3D array with three layers: blue, green, and red layers, each of which has a value between 0 and 255. The reading of the matrix then occurs, for which the programme picks a smaller picture, referred to as the “filter,” from which the information (or kernel) is read. There is no difference between the depth of the filter and the depth of the input. The filter then generates a convolution movement that moves together with the input picture, moving one unit to the right of the image every time it is used.

After that, it multiplies the values by the values of the original image. Each multiplied figure is added together, and a single number is formed as a result of this process. Iterating the method with the full picture results in a matrix that is smaller than the original input image.

The feature map of an activation map is the last array in the process of creating an activation map. In order to conduct operations such as edge detection, sharpening, and blurring, it is necessary to convolute a picture by applying several filters. All that required is the specification of parameters such as the size of the filter, the number of filters, and/or the network’s architectural design.

From a human standpoint, this behaviour is analogous to recognising the basic colours and edges of a picture. However, in order to identify the picture and detect the traits that distinguish it as, for example, that of an elephant and not that of a cat, distinguishing characteristics such as the elephant’s enormous ears and trunk must be recognised. In this case, the nonlinear and pooling layers will be used to help.

The nonlinear layer (ReLU) is added after the convolution layer, and it is responsible for increasing the nonlinearity of the picture by applying an activation function to the feature maps. The ReLU layer eliminates any negative values from the picture and boosts the image’s correctness. Despite the fact that there are various procedures available, such as tanh or sigmoid, ReLU is the most common since it can train the network much more quickly.

In the next stage, many photos of the same item are created so that the network can always identify the image, regardless of its size or position on the network. For example, in the elephant image, the network must be able to

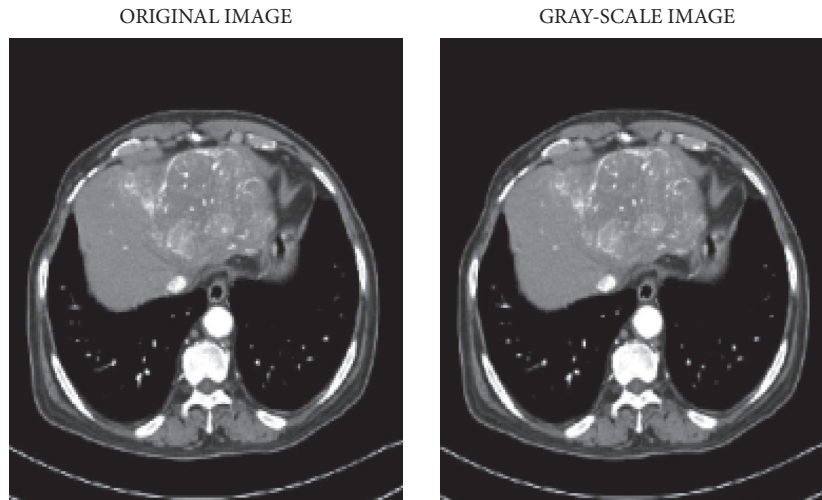


FIGURE 6: Image outputs in preprocessing.

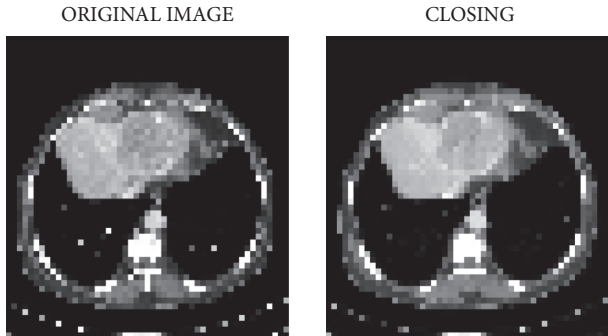


FIGURE 7: CT scan.

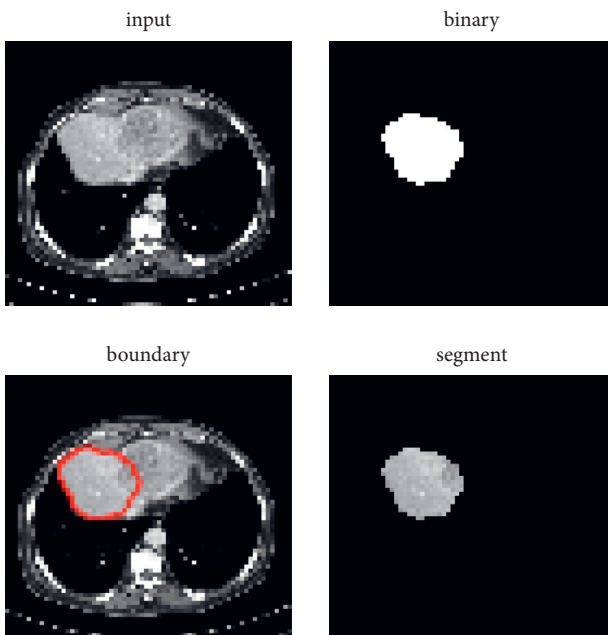


FIGURE 8: Segmented CT scan image.

detect the elephant regardless of whether it is walking, standing still, or racing. It is necessary to have picture flexibility, and here is where the pooling layer is useful.

It works in conjunction with the picture's dimensions (height and width) to gradually shrink the size of the input image, allowing the items in the image to be seen and identified no matter where they are positioned in the image space.

Pooling also aids in the prevention of "overfitting," which occurs when there is too much information and no room for new ones. Max pooling is perhaps the most well-known example of pooling, in which the picture is split into a succession of nonoverlapping sections.

Max pooling is the process of detecting the maximum value in each region of the picture in order to eliminate any unnecessary information and reduce the size of the image to its smallest possible size. It also helps to account for distortions in the picture as a result of this activity.

The fully connected layer is the next step, which includes an artificial neural network for use with CNN. It is possible to forecast the picture classes with improved accuracy by using an artificial network that incorporates diverse information. At this point, the gradient of the error function is computed in relation to the weight of the neural network being considered. The weights and feature detectors are tweaked to get the best possible performance, and the process is performed over and over again.

The classification process is performed using the method of convolutional neural network. Convolutional neural network consists of many layers, which would give the certain rate of classification in the three categorised database images. Appendix 1 represents the flowchart of the proposed work. This would help the patient and the practitioners to identify the early stage of liver cancer and help with the diagnosis. Figure 9 shows the classification results of the proposed dataset.

Table 2 represents the performance metrics of the proposed work with various sample images.

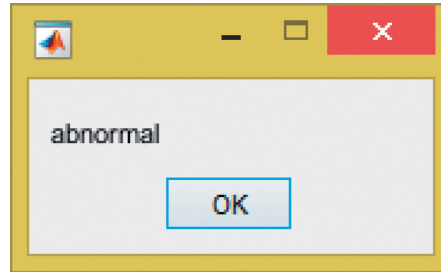


FIGURE 9: Classification of CT scan.

TABLE 2: Performance metrics.

	ACCU	SENS	SPECIFI	FPR	PPV	NPV
CT1	99.28919	100	99.27504	26.69522	73.30478	100
CT2	99.42793	100	99.41669	22.8866	77.1134	100
CT3	99.17108	100	99.15489	30.21232	69.78768	100
CT4	99.13311	100	99.1153	30.09321	69.90679	100
CT5	99.07229	100	99.05211	30.35376	69.64624	100
CT7	99.26252	100	99.24729	26.70654	73.29346	100
CT8	93.36848	100	99.35712	26.33125	73.66875	100
CT9	91.3963	100	99.38447	23.89629	76.10371	100
CT10	90.35506	100	99.34252	25.27013	74.72987	100
CT11	93.37441	100	99.36253	25.13465	74.86535	100
CT12	95.33381	100	99.32114	26.30273	73.69727	100
CT13	93.34468	100	99.33146	24.88874	75.11126	100
CT14	94.28268	100	99.269	27.70199	72.29801	100
CT15	90.29426	100	99.28084	27.43989	72.56011	100

\* ACCU: accuracy; SENS: sensitivity; SPECIFI: specificity; FPR: false-positive rate; PPV: positive prediction value; NPV: negative prediction value; ROC: receiver operating characteristic.

TABLE 3: Comparison of the proposed work.

Classifiers	Accuracy (%)	Precision (%)	F1 score (%)	ROC curve (%)
SVM [23]	98.11	99	98.3	99.62
Naive Bayes [24]	98.11	98.1	99.3	97.24
CNN [25]	98.11	97.9	99.5	97.07
Proposed	98.8	99	99.3	98.44

Table 2 represents the various CT images performance metrics using the proposed work.

Table 3 represents the comparison of the proposed work with the existing work. The SMRF method is implemented in the Hadoop cluster distributed computing environment. We use the Weka workbench to run classic random forest with the same settings as before, and we set the  $K$  value to 100 to be able to compare the accuracy levels of the two methods side by side. As an assessment measure, we employ 10-fold cross-validation to evaluate the results of various approaches. As a result, we compute the mean of the accuracy of these two classifiers in order to decrease the bias of datasets that have been classified in a certain way.

## 5. Conclusion

The SMRF algorithm yields the better results than the traditional algorithm in the case of liver cancer prediction. This proposed model has developed based on the MapReduce

model. This made the drastic changes in the big data analysis or in cloud computing environment. The comparative study with the various algorithms gives the better results of the implemented results. The proposed structure is based on the decision trees, which is used on the drug discovery of the liver cancer. To draw a conclusion that the SMRF algorithm is more suitable to classify massive datasets in distributing computing environment than the traditional random forest algorithm [30].

## Appendix

### A. Pseudocode of SMRF Algorithm

SMRF algorithm.

- (1) Map;  $V_i \in (1, 2, 3, \dots, \text{data})$
- (2) Input: Set of training dataset  $D$ , corresponding the attribute set  $M$ , randomly picked the subset of attributes  $m$  per tree.

- (3) Output: Decision trees generated by IG
- (4) Negotiate the scale of the Random Forest  $K$  parameter in computer computing environment clusters or cloud
- (5) Initialize dataset, generate bootstrap samples by Bagging algorithm
- (6) Build tree per bootstrap sample, randomly pick a subset of attributes  
While  $j5 > (x_i, y_j)$   
do
- (7) For each candidate attribute IG
- (8) Calculate the Max (IG) = argmax IG; Splitting on Max (IG) attribute;
- (9) End

## Data Availability

The data that support the findings of this study are available on request from the corresponding author.

## Conflicts of Interest

All authors declare that they do not have any conflicts of interest.

## Acknowledgments

The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through a research group program under grant number R. G. P. 1/399/42.

## References

- [1] J. A. DiMasi, R. W. Hansen, and H. G. Grabowski, "The price of innovation: new estimates of drug development costs," *Journal of Health Economics*, vol. 22, no. 2, pp. 151–185, 2003.
- [2] J. J. Irwin, T. Sterling, M. M. Mysinger, E. S. Bolstad, and R. G. Coleman, "Zinc: a free tool to discover chemistry for biology," *Journal of Chemical Information and Modeling*, vol. 52, no. 7, pp. 1757–1768, 2012.
- [3] W. P. Walters, M. T. Stahl, and M. A. Murcko, "Virtual screening—an overview," *Drug Discovery Today*, vol. 3, no. 4, pp. 160–178, 1998.
- [4] M. K. Ahirwar, P. K. Shukla, and R. Singhai, "CBO-IE: a data mining approach for healthcare IoT dataset using chaotic biogeography-based optimization and information entropy," *Scientific Programming*, vol. 2021, Article ID 8715668, 14 pages, 2021.
- [5] M. H. Forouzanfar, M. D. Forouzanfar, K. J. Foreman et al., "Breast and cervical cancer in 187 countries between 1980 and 2010: a systematic analysis," *The Lancet*, vol. 378, no. 9801, pp. 1461–1484, 2011.
- [6] "The Apache Mahout Project," <http://mahout.apache.org/>.
- [7] J. Dean and S. Ghemawat, "MapReduce," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008.
- [8] V. K. Trivedi, P. Shukla, and A. Pandey, "Plant leaves disease classification using bayesian regularization Back propagation deep neural network," *Journal of Physics: Conference Series*, vol. 1998, no. 1, Article ID 012025, 2021.
- [9] A. S. Alghamdi, K. Polat, A. Alghoson, A. A. Alshdadi, and A. A. Abd El-Latif, "A novel blood pressure estimation method based on the classification of oscillometric waveforms using machine-learning methods," *Applied Acoustics*, vol. 164, Article ID 107279, 2020.
- [10] S. A. K. R. Sherif and L. I. U. Anna, "FAYOUMI, king abdulaziz university, the family of MapReduce and large-scale data processing systems," *ACM Computing Surveys*, vol. 46, no. 1, 2013.
- [11] N. K. Rathore, N. K. Jain, P. K. Shukla, U. Rawat, and D. Rachana, "Image forgery detection using singular value decomposition with some attacks," *National Academy Science Letters*, vol. 44, pp. 331–338, 2021.
- [12] A. S. Alghamdi, K. Polat, A. Alghoson, A. A. Alshdadi, and A. A. Abd El-Latif, "Gaussian process regression (GPR) based non-invasive continuous blood pressure prediction method from cuff oscillometric signals," *Applied Acoustics*, vol. 164, Article ID 107256, 2020.
- [13] B. Abd-El-Atty, A. M. Iliyasu, H. Alaskar, and A. A. Abd El-Latif, "A robust quasi-quantum walks-based steganography protocol for secure transmission of images on cloud-based E-healthcare platforms," *Sensors*, vol. 20, no. 11, p. 3108, 2020.
- [14] G. Khambra and P. Shukla, "Novel machine learning applications on fly ash based concrete: an overview," *Materials Today Proceedings*, pp. 2214–7853, 2021.
- [15] "Apache. hadoop documentation," <http://hadoop.apache.org/core>.
- [16] M. Hammad, M. H. Alkinani, B. B. Gupta, and A. E. L. Ahmad, "Myocardial infarction detection based on deep neural network on imbalanced data," *Multimedia Systems*, Springer, Berlin, Germany, 2021.
- [17] P. K. Shukla, J. K. Sandhu, A. Ahirwar, D. Ghai, P. Maheshwary, and P. K. Shukla, "Multiobjective genetic algorithm and convolutional neural network based COVID-19 identification in chest X-ray images," *Mathematical Problems in Engineering*, vol. 2021, Article ID 7804540, 9 pages, 2021.
- [18] M. Hammad, A. M. Iliyasu, A. Subasi, E. S. L. Ho, and A. A. A. El-Latif, "A multitier deep learning model for arrhythmia detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, 2021.
- [19] A. Gupta, *Learning Apache Mahout Classification*, p. 68, Packt Publishing, Birmingham, United Kingdom, 2015.
- [20] O. Trott and A. J. Olson, "Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading," *Journal of Computational Chemistry*, vol. 31, no. 2, pp. 455–461, 2010.
- [21] E. Glaab, *Building a Virtual Ligand Screening Pipeline Using Free Software: A Survey*, Briefings in Bioinformatics, Oxford, UK, 2015.
- [22] J. Meiler and D. Baker, "ROSETTALIGAND: protein-small molecule docking with full side-chain flexibility," *Proteins: Structure, Function, and Bioinformatics*, vol. 65, no. 3, pp. 538–548, 2006.
- [23] S. Hafizah and A. Ubaidillah, "Cancer detection using artificial neural network and support vector machine," *A Comparative Study Journal Teknologi*, vol. 65, no. 1, pp. 73–81, 2013.
- [24] G. Ilakkiya and B. Jayanthi, "Liver cancer classification using principal component analysis and fuzzy neural network," *International Journal of Engineering Research and Technology*, vol. 10, no. 2, 2013.
- [25] S. Pandit, P. K. Shukla, A. Tiwari, P. K. Shukla, and R. Dubey, "Review of video compression techniques based on fractal

- transform function and swarm intelligence,” *International Journal of Modern Physics B*, vol. 34, no. 8, Article ID 2050061, 2020.
- [26] A. Kumar and C. J. Venkateswaran, “Estimating the surveillance of liver disorder using classification algorithms,” *International Journal of Computer Application*, vol. 57, no. 6, 2012.
- [27] A. Sedik, M. Hammad, F. E. Abd El-Samie, B. J. Birj, and A. A. E. L. Ahmad, “Efficient deep learning approach for augmented detection of coronavirus disease,” *Neural Comput & Applic*, 2021.
- [28] H. R. Kiruba and G. Tholkappiaarasu, “An intelligent agent based framework for liver disorder diagnosis using artificial intelligence techniques,” *Journal of Theoretical and Applied Information Technology*, vol. 69, no. 1, 2014.
- [29] S. Dhamodharan, “Liver disease prediction using bayesian classification,” in *Proceedings of the 4th National Conference on Advanced Computing, Applications & Technologies*, Special Issue, Rohtak, India, February 2014.
- [30] V. Roy, P. K. Shukla, A. K. Gupta, V. Goel, P. K. Shukla, and S. Shukla, “Taxonomy on EEG artifacts removal methods, issues, and healthcare applications,” *Journal of Organizational and End User Computing*, vol. 33, no. 1, pp. 19–46, 2021.

## Research Article

# Biosensor-Assisted Method for Abdominal Syndrome Classification Using Machine Learning Algorithm

Charu Gandhi,<sup>1</sup> Sayed Sayeed Ahmad ,<sup>2</sup> Abolfazl Mehbodniya ,<sup>3</sup> Julian L. Webber ,<sup>4</sup> S. Hemalatha,<sup>5</sup> Haitham Elwahsh,<sup>6</sup> and Basant Tiwari <sup>7</sup>

<sup>1</sup>Department of CSE & IT, Jaypee Institute of Information Technology, Noida, India

<sup>2</sup>College of Engineering and Computing, Al Ghurair University, Dubai, UAE

<sup>3</sup>Department of Electronics and Communication Engineering, Kuwait College of Science and Technology (KCST), Kuwait, Kuwait

<sup>4</sup>Graduate School of Engineering Science, Osaka University, Osaka, Japan

<sup>5</sup>Department of Computer Science and Engineering, Panimalar Institute of Technology, Chennai, Tamil Nadu, India

<sup>6</sup>Faculty of Computers and Information, Kafrelsheikh University, Kafrelsheikh, Egypt

<sup>7</sup>Department of Computer Science, Hawassa University, Awasa, Ethiopia

Correspondence should be addressed to Basant Tiwari; [basanttiw@hu.edu.et](mailto:basanttiw@hu.edu.et)

Received 3 November 2021; Revised 21 December 2021; Accepted 28 December 2021; Published 28 January 2022

Academic Editor: Mohammed A. A. Al qaness

Copyright © 2022 Charu Gandhi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The digestive system is one of the essential systems in human physiology where the stomach has a significant part to play with its accessories like the esophagus, duodenum, small intestines, and large intestinal tract. Many individuals across the globe suffer from gastric dysrhythmia in combination with dyspepsia (improper digestion), unexplained nausea (feeling), vomiting, abdominal discomfort, ulcer of the stomach, and gastroesophageal reflux illnesses. Some of the techniques used to identify anomalies include clinical analysis, endoscopy, electrogastrogram, and imaging. Electrogastragram is the registration of electrical impulses that pass through the stomach muscles and regulate the contraction of the muscle. The electrode senses the electrical impulses from the stomach muscles, and the electrogastragram is recorded. A computer analyzes the captured electrogastragram (EGG) signals. The usual electric rhythm produces an enhanced current in the typical stomach muscle after a meal. Postmeal electrical rhythm is abnormal in those with stomach muscles or nerve anomalies. This study considers EGG of ordinary individuals, bradycardia, dyspepsia, nausea, tachycardia, ulcer, and vomiting for analysis. Data are collected in collaboration with the doctor for preprandial and postprandial conditions for people with diseases and everyday individuals. In CWT with a genetic algorithm, db4 is utilized to obtain an EGG signal wave pattern in a 3D plot using MATLAB. The figure shows that the existence of the peak reflects the EGG signal cycle. The number of present peaks categorizes EGG. Adaptive Resonance Classifier Network (ARCN) is utilized to identify EGG signals as normal or abnormal subjects, depending on the parameter of alertness ( $\mu$ ). This study may be used as a medical tool to diagnose digestive system problems before proposing invasive treatments. Accuracy of the proposed work comes up with 95.45%, and sensitivity and specificity range is added as 92.45% and 87.12%.

## 1. Introduction

Human physiology comprises the nervous system, cardiovascular system, respiratory system, and digestive system. The digestive system, among these systems, is one of the most powerful systems where the stomach plays a vital part with its accessories such as the esophagus, duodenum, small intestines, and large intestines. The digestive system consists of the gastrointestinal tract, the mouth twisting pipe to the

anus, and other organizations that assist the body to break down and absorb food. Food and drinks must be transformed into smaller molecules of nutrients before they can be absorbed and transported to cells throughout the body for food. Digestion is when food and beverage are divided into smaller pieces to allow the body to utilize them to construct and feed the cells and provide energy. However, worldwide, many individuals have stomach illnesses linked with gastric motility abnormalities, such as dyspepsia (improper

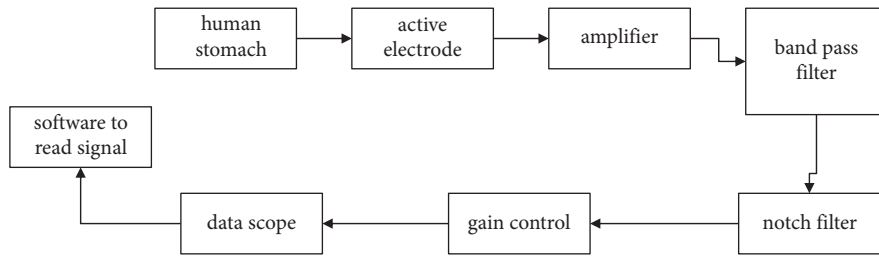


FIGURE 1: EGG recording procedure.

digestion), inexplicable nausea (sensation failure), vomiting, abdominal pain, stomach ulcer, and gastroesophageal reflux disorders. Some of the techniques used to identify anomalies include clinical analysis, endoscopy, electrogastrogram, and imaging. Of the abovementioned techniques, electrogastrogram (EGG) is noninvasive and cost-effective [1]. Electrogastrogram is the registration of electrical impulses that pass through the stomach muscles and regulate the contraction of the muscle. The electrode senses electrical impulses from the stomach muscles, and the EGG is recorded to investigate digestive system problems. The research is carried out with diseases such as bradygastria, dyspepsia, nausea, tachygastria, ulcers, and vomiting. EGG data are evaluated using statistical parameters, method of wavelet transformation, and approach to the neural network.

*1.1. Digestive System.* The human body has to survive with food. However, the food in the body is just a tiny component of the process. The meal must be divided into chemical substances that the body can utilize. This entire process is known as digestion. As shown in Figure 1, the digestive system comprises several organ breakdowns and disposal of chemical ingredients of meals, including stomach, pancreas, liver, gallbladder, small gut, and considerable intestinal fluctuation. Digestion in the mouth starts. The chewed meal is lubricated and humidified by saliva, a watery mucus, and enzyme composition. The second phase of digestion takes place in the stomach, in which stomach fluids are separated and combined with the meal for fluid termed chyme. It moves from the stomach into the duodenum, where the liver and pancreas handle more enzymes. The liver creates bile to break down fats stored in the gallbladder and goes into the duodenum via the bile duct. The pancreas generates and releases enzymes for the digestion of proteins and carbohydrates. After treating the pancreatic enzymes and bile, chyme passes via the small intestine. The small intestine is treated with some extra enzymes produced by the intestinal wall, and the digestive process is over. Absorption occurs in the small gut. In the large intestine, water is eliminated. Digestion performs an essential function in the stomach. It appears like a flattened ball when it is empty, but when filled, it can contain approximately two-quarters of food and drink for 1 foot and 6-inch width. The stomach comprises chemical and mechanical activity. Several substances in the stomach, notably pepsin, rennin, and lipase digestive enzymes, combine to break up food. In addition, hydrochloric acid provides an appropriate environment for

the enzymes and supports digestion. The watery mucus protects the muscle walls of the stomach from the digestion of the acid or enzymes. The mechanical activity of the stomach muscles constraints and relaxes in a continual mixture, whipping and churning the stomach's substance in the chyme.

The EGG is comparable to the cardiovascular electrocardiogram (ECG). It captures electric impulses that pass through the stomach muscles that regulate the muscle's contractions. It also monitors the activity of the stomach wall before and after the intake of meals. EGG traces have a 3-cycle frequency of sinusoidal waveforms per minute. Clinical investigations have demonstrated a good relationship between these cutaneous recordings and those obtained by serially implanted electrodes. EGG is assessed as a regular electric rhythm produced by stomach muscles in ordinary people, and the strength (voltage) of the electric current rises after food, and in patients with stomach muscle disorders, the rhythm is erratic, or electric power increases after a meal [2, 3]. There are no adverse effects of EGG recording, and the research is painless.

EGG is presently used in research and clinical settings since it is an efficient technique for stomach electrophysiology and gastric motility disorders pathophysiological studies [4]. Since the first EGG recording development was particularly sluggish in this area compared with other electrophysiological cutaneous measures because of its difficulties in acquiring data, there is a lack of knowledge of EGG at unique frequency and amplitude. Parameters are being discussed, and EGG's clinical use is still being studied [5].

*1.2. The Motivation of the Proposed Work.* Therefore, a frequency range must be established for ordinary individuals and dysrhythmias, EGG recording standardization, and sophisticated analytical techniques for extracting and interpreting quantitative EGG data. Additional EGG result studies are also required to establish the use of EGG in the therapeutic context. Today, doctors and biological researchers are interested in the quantitative analysis of EGG. Currently, electrogastrogram application in India is not performed up to date since it is a completely noninvasive method to study digestive problems. Many researchers follow this subject to get accurate findings. Acquisition and analysis of EGG to help the physician diagnose digestive system problems at an early stage with considerable accuracy.

## 2. Background Analysis

The author's limb records the EGG of a five-week-old kid with pyloric stenosis. The EGG seemed comparable to an electrocardiogram (ECG) with progressively shifting baselines. Smout et al. demonstrated that when contractions occur, the amplitude of the EGG rises [6]. Chen et al. have developed a new method of spectrum analysis based on an adaptive, moving average model [7]. This technique gave better precision and more accurate information regarding frequency changes in electric stomach activity. It is beneficial for identifying short-term dysrhythmic occurrences of stomach activity.

*2.1. Analysis of Electrogastrogram.* The first to use the spectrum analysis method to EGG was Stevens and Worrall and then to analyze EGG data using Fourier transform [8]. Chen et al. conducted arithmetical analyses to examine the change in EGG designs between individual subject collections [9]. *T*-tests were performed for students in pairs and unpaired to evaluate differences and assign statistical significance. Hrair Simonian et al. (2004) stated the definition of average values as mean  $\pm 2$  [10]. The standard abnormal deviation is calculated when one of the values above is beyond the range. Ding et al. employed an electrogastrography to detect slow stomach waves, and the authors developed a multiresolution technique to deconstruct the EGG signal using the Daubechies wavelet function [11]. Zhenghu has created a novel wavelet-based treatment technique of EGG signals with an excellent application viewpoint. It is easy and quick to produce accurate charts and frequency identification features for refining [12]. Kania et al. have investigated the significance of the proper selection of mom's wavelet for decomposing the ECG signal noise [13]. The researchers concluded they got a high-quality signal on the first and fourth degradation levels for the wavelet db1 and sym3 for the fourth degradation level.

Zhongjia et al. utilized the Social Sciences Statistical (SPSS) package for the examination of alteration (ANOVA) and range analysis to show in what way various pressure values and moisture content affect the forming density and to assess the significance of the two variables to shape biofuel density [14]. Xing-ce et al. conducted a brain vascular segmentation preprocessing technique based on the parametric statistical model [15]. After the preprocessing stage, the writers examined the brain picture as an input to the parameterized statistical model and secreted the tiny branches of their brain vessels.

Elsayad used neural Learning Vector Quantization (LVQ) networks to identify ECG dataset arrhythmias [16]. For a future study on any biosignals, the experimental findings suggest LVQ algorithms. Ramanathan et al. have categorized the lung sound using the ANN architecture wavelet coefficients [17]. For the standard or pathological categorization of EGG signals, Tsung-Nan Lin et al. observed the optimum outcome by having six to seven concealed neurons in the network [18]. Slow convergence and key user-dependent factors are barriers to distinguishing normal and

aberrant signals. Dutta et al. utilized a heartbeat diagnostic medical diagnostic tool to accomplish an accurate and timely diagnosis of heart arrhythmia to provide a patient who uses an extractor combined with an artificial neural network (ANN) classification with adequate medical care [19]. Based on a sliding dot product method, the attribute filter uses the frequency-domain data of cross-spectrum. Ahsan et al. presented the method of identifying various hands for complicated EMG-based classification and pattern recognition applications with the aid of an artificial neural network (ANN) [20]. The authors utilized BPN with the training method Levenberg-Marquardt for gesture identification. Shiau et al. conducted a cardiac gesture examination for all serial and patient pictures utilizing the BPN network analysis [21]. Barrea and Barrea utilized a local fuzzy c-means clustering to analyze the spectroscopic data to verify a novel medication for prostate malignancy called clioquinol (CQ) [22]. To address the drawbacks of this current work, the work suggested an experimental set-up for recording electrogastrogram cutaneously with patients and studying digestive systems diseases to see the deviation of stomach signal from average to abnormal. To build an EGG database containing bradygastria diseases, dyspepsia, nausea, tachygastria, ulcer, and vomiting disorders along with the normal subject, Naive Bayesian Classification to conduct statistical analysis and categorize the EGG signals based on higher-order momentum (NBC), and Continuous Wavelet Transform (CWT) applications to EGG signals for disorder identification and categorization, they proposed the appropriate neural network and training method using an unsupervised neural network of adaptive resonance theory (ART-NN) (LVQ) and analyze the performance of BPNN for the nine training algorithms, i.e., gradient descent with momentum and adjustable learning rate, backing sprouting (GDX), resilient back spraying (RP), conjugate back-propagation gradient with updates from Fletcher-Reeves (GCF), a combination of backpropagation gradient with updates from Polak-Ribiere (CGP), the connection of backpropagation gradient with restarts from Powell-Beale (CGB), and scaled juxtaposition to use the clustering method fuzzy c-means to discrimination between standard and arrhythmic EGG signals to enhance classification efficiency. Table 1 shows the various comparison rate of the existing methodology.

The network was trained on two classes of EGG signals and used to predict 102 samples in the patients. An average EGG signal (normogastria) was identified as one of two signal classes in all of the investigations described above, with the aberrant EGG signal designated as the other. For this study, we will treat the states of normogastria, bradygastria, and tachygastria independently to evaluate two- and three-class problems. The electrogastrography signals are divided into three groups for three-class signal classification and two groups for two-class signal classification, respectively (normal and abnormal). We successfully classify two-class and three-class EGG signals using statistical machine learning algorithms such as Logistic Regression, SVM, and KNN. We compare their performances to suggest which applied ML algorithm is the most effective for a multiclass



TABLE 1: Comparison of Existing works.

Authors	Stomach position	Electrode position	Coordinates	Disadvantage
[19]	Cutaneous reference points	Proximal electrodes	Placed in the costal margin of the stomach	Commonly used but time consumption
[20]	Sonography method	2 electrodes	Left side of the abdomen and ventral region	Overuse of electrodes may lead to severe problems in accuracy rate
[21]	Cutaneous reference points method	1 electrode	Between umbilicus and xiphoid process	Low signal to noise ratio
[22]	X-ray method	6 electrodes	At least in one channel of the abdomen	Low signal to noise ratio
[23]	Cutaneous reference points method	5 electrodes	Umbilicus and xiphoid process	Pick up propagation is too low

EKG signal classification. The total accuracy, F1 score, precision, and recall are the performance metrics taken into consideration in this research.

### 3. Proposed Work

Most individuals throughout the globe suffer health issues, mainly owing to dietary intake and digestive system diseases. Today, the Endoscope technique is used to study the issues of a laborious, costly, and intrusive approach to digestive system disease. To detect the electric signal cutaneously from the stomach, a noninvasive, inexpensive, and painless electrogastrogram (EGG) method has been developed as an initial method to investigate gastric disorders before encouraging the Endoscope procedure for noninvasive gastric disease and benign tumors. Because of its non-invasive nature and the recent improvements in EGG recording and computerized analysis methods, EGG has become a valuable researcher’s tool for studying the electrophysiology of gastric motility disorders and is now used in both research and therapeutic settings. Contrary to ECG, EEG, and EMG, electrogastrogram signal in any database is not accessible. This study aims to develop a technique for obtaining high-quality EGG noise-free signal, cheap cost, and reduced complexity of purchases for any individual with or without symptoms for preliminary examination of digestive system problems before invasive processes.

The stomach may be monitored closely, intraluminally, or cutaneously by gastric myoelectric activity (GMA). The serosal record may be achieved by surgically inserting electrodes on the serosal surface of the stomach. The intraluminal record may be obtained by incubating a catheter with electrodes. Suction is typically used to provide good contact between the electrodes and the mucosal wall of the stomach. The serosal and intraluminal elements may record slow waves and spikes because they constitute a limited number of smooth muscle cells with myoelectric activity. These techniques are intrusive, and their applicability in animals and laboratory environments is restricted. EGG, a cutaneous measurement of GMA by surface electrodes, is extensively utilized in people and clinical settings since it is noninvasive and does not disrupt continuous stomach

TABLE 2: Categorization of dataset.

EGG	Mean Age (years)	Male	Female
Normal	33	32	18
Bradygastria	28	27	23
Dyspepsia	38	29	21
Nausea	45	28	22
Tachygastria	36	26	24
Ulcer	34	31	19
Vomiting	35	19	31

action [23]. Various validation studies have shown the accuracy of the EGG by comparing it to the mucosal and serosal electrode record [24]. Reproducibility of the EGG recording, without substantial daily fluctuations, was shown. In adults, gender and age seem to have little effect on the EGG. The recording is done using surface electrodes noninvasively.

One of the primary goals was to categorize electrogastrography signals with high accuracy under both a two-class and a three-class classification model, which was accomplished. The electrogastrography signals are divided into three groups for three-class signal classification (normogastria, bradygastria, and tachygastria) and two groups for two-class signal classification (normogastria, bradygastria, and tachygastria) (normal and abnormal). Our machine learning techniques for classifying two and three classes of EGG signals included Logistic Regression, SVM, and KNN, which were all implemented. As a consequence of our experiments, we find that the SVM method is the most accurate in classifying the two- and three-class signals, with accuracy rates of 100 percent and 92.11 percent, respectively. The SVM algorithm attained a maximum F1 score, precision, and recall value of 100 percent and 92 percent, respectively, for the two and three classes of EGG signals, using the SVM technique. In addition, SVM effectively identified EGG signals with a high degree of separability of 100 percent and 92 percent in the two-class and three-class classifications, respectively, with a high degree of separability of 100 percent and 92 percent. Following our investigation, we have concluded that SVM can be effectively used to reliably classify multiclass EGG signals and forecast the signal features of such signals.

### 3.1. Materials and Methods

**3.1.1. Image Acquisition.** By the Helsinki declaration [25], EGG data acquisition shall be carried out by explaining the procedure of all subjects (normal and abnormal) under the guidance and monitoring of renowned hospital gastroenterologists who have been admitted to participate in the study. More than a thousand sample EGGs, including patients and ordinary individuals from both male and female categories of various age groups, were included in this recording, as indicated in Table 2 [26–28]. Electrogonogram is recorded with a minimum length of half an hour under preprandial (before meal) and postprandial (after food) conditions.

Figure 1 shows the electrogastrogram recording set-up utilizing Method E. Active electrodes hit the abdominal bioindication through motility on the skin. The electrodes' output consists of input into the SCU, an instrument amplifier, a passband filter, a band-reject filter, and gain control. SCU contains the process such as adaptation to the range, filtering, conversion, amplification, insulation, and other operations required for subsequent sensors processing. A device amplifier is employed in SCU for improving the electrode sensing potential. With the aid of a voltage signal as input, an amplifier produces a signal of linear variant at the output. It is a PLL amplifier, typically with high input impedance, short drift, and increased frequency rejection. A *filter* is an instrument that transmits frequencies in a definite range and limits frequencies beyond this scope. A band-reject filter is also termed a Notch filter employed to remove a specific frequency segment of a signal. Also, it can be employed in minimizing or avoiding feedback. In many electrical devices, the control of gain is an adaptive mechanism that averages the signal level at the output, thereby adjusting the gain to a suitable level for different input signal levels, conditioning the signals almost for all data with normalized sensor signals filtered to appropriate levels compatible with digital-analog conversions into computer processor recording and analysis.

Approximately 1000 individuals were examined based on the gastroenterologist's previous knowledge. The EGG database comprises about 500 individuals, with an average of 70 per category of 6 diseases and a normal category based on the conditional probability based on Baye's theory. The Sri Ramakrishna Hospital was guided to capture EGG signals at the Department of Biomedical Engineering, Sri Ramakrishna Engineering College, Coimbatore. Initially, the sample collected from MEDINDIA Hospital and PSG Hospital, Coimbatore, generated the database. The average and pathological thresholds are established following the physician from the database. This was utilized as the benchmark or the fundamental truth. The recording stability has been confirmed in PSG Hospitals and MEDINDIA Hospitals, Coimbatore. 60 samples per minute of EGG data were utilized for normal and disorder patients as classification inputs.

**3.1.2. Feature Extraction with Optimization.** Continuous Wavelet Transformation (CWT) has been created as an

(1,1)	(1,2)	(1,3)	(1,4)
(2,1)	(2,2)	(2,3)	(2,4)
(3,1)	(3,2)	(3,3)	(3,4)
(4,1)	(4,2)	(4,3)	(4,4)

FIGURE 2: Horizontal value of the input signal.

TABLE 3: Feature Extraction of proposed work.

Feature number	Input features
1, 2	Second rotational momentum
3, 4	Dissimilarity
5, 6	Correlation \variance
7, 8	Moment of reverse difference
9, 10	The average sum
11, 12	Their variance
13, 14	Entropy sum
15, 16	Entropy
17, 18	Variance of difference
19, 20	Entropy difference
21, 22	Correlation information measure I
23, 24	Correlation information measure II
25, 26	The maximum coefficient of correlation
27, 28	Second angular moment

alternate FT method to decrease the complexity of data extraction. The word wavelet refers to a bit of wave. The smallness refers to the condition of a finite length of this function. The wave refers to the state of oscillation. A feature (wavelet) multiplies the signal to obtain the CWT of a signal, and the transform is calculated independently for various parts of the time domain signal.

Four subbands result in a single wavelet decomposition of ROI. Daubechies wavelet filter [29] is used for decomposition in order 2. The subband, which displays the changes with the most apparent appearance between various textures, has the most significant difference in the histogram. This subband is selected to be further processed.

Haralick et al.'s method to spatial dependence matrix in grey color (1973) is a well-known statistical technique for collecting secondary texture data from images used for this study [30]. The SGLDM comprises the assessment of the discrete second-order function.

The collection of all horizontally adjacent resolution cells separated by distance is shown in Figure 2. Figure 2(a) displays a grey image of four to four of 0 to 3 tones. Figure 2(b) shows a grey-tone spatial dependence matrix (b). For example, at (2, 1) distance 1 of the horizontal PH matrix, the total number of times two grey color tones, with values of 2 and 1, occurred horizontally adjacent to each other. We count the number of cell resolution pairs in RH for calculating this number, such that the first resolution of the pair has a grey ton 2 and the second resolution of the pair's cell is grey ton 1. Figures 2(c)–2(f) show spatial addition matrices of all four distances of one grey tone.

The coincidence conditions are determined for offset angles of  $0^\circ$  to  $135^\circ$  with a step value of  $45^\circ$  for a certain

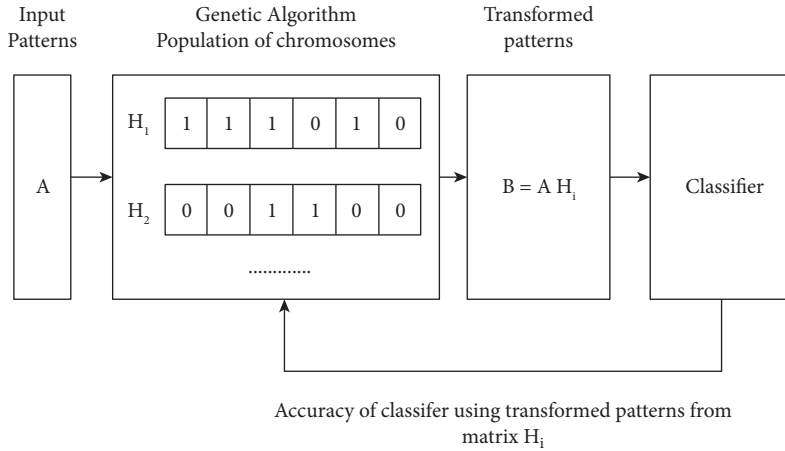


FIGURE 3: Feature extraction using genetic algorithm optimization with continuous wavelet transform.

offset distance. 14 SGLDM measurements are generated from each of these matrices. Each measure is characterized by the mean and the range (difference between maximum and minimum) across four offset angles; this results in 28 characteristics for, respectively, offset distance (Table 3).

Features for offset distances of 1, 2, 3, and 4 are computed. This gives a total of 112 characteristics. The second corner feature reflects the consistency of the textures. Contrast quantifies the number of local differences in a picture. Correlation is a grey-tone linear picture dependency metric. Variance is a measure of distribution dispersion. The contrary difference currently is an amount of local double uniformity. Entropy quantifies the grey level distribution unpredictability. Although the 14 features include information on the textural properties of the picture, it is difficult to determine the exact textural properties of each feature.

The wave energy characteristic represents the energy distribution across size and direction on the frequency axis and has been extremely effective for characterizing texture [31].  $x(m, n)$  represents the subband numbers with the extreme histogram alteration of  $1/m/M$  and also  $1/n/N$ .

**3.1.3. Feature Selection.** *Feature selection* is a method of selecting relevant characteristics to provide an effective and better solution to a particular issue. Ideally, an optimal subset of characteristics should be selected from the set of accessible features required and adequate for addressing the issue. Selection of features is essential since not all features offered are helpful. Some functions may be duplicated, while others during the learning period may create confusion. This increases the complexity of the function room needlessly, which in turn takes more computer time to learn or to find a solution to the provided issue.

*(1) Genetic Algorithms in Feature Selection and Weighting.* GAs are similar, iterative aspect optimizers that have successfully been used to a wide range of optimization issues, including numerous pattern recognition and classification applications. GA is inherently suited to selecting features since the issue has an exponential search area. Siedlecki and

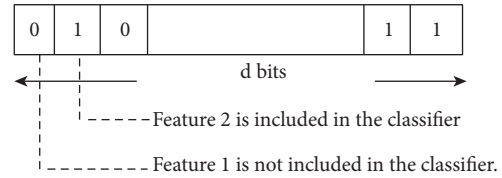


FIGURE 4: Single binary vectorization in genetic algorithm.

Sklansky’s pioneering work (1989) showed that GA was superior to classical representative algorithms [32]. Much research was subsequently produced which showed the benefits of the GAs for feature selection [33–36]. As an optimization issue, the problem of dimensional decrease is healthy suitable for preparation.

GA aims at finding a transformed pattern in the  $d$ -dimensional input patterns in an  $m$  ( $m < d$ ) dimension that optimizes the set of optimization criteria. In general, altered designs are assessed with their dimensionality and classification accuracy, rarely accomplished using a particular classifier in the patterns. Figure 3 shows the design of an extractor based on the accuracy of GA classification as an assessment criterion. It holds a population with competitor matrices. Each matrix in this population is estimated by multiplying the input patterns and creating a sequence of changing patterns later, which are submitted to the classifier that segments the patterns into a training netlist for classifier learning and a test pattern for the accuracy of classification. To evaluate the quality of the transformation matrix used for collecting modified designs, the obtained precision is then returned to GA. The GA utilizes this information to seek a change that reduces the size of changing patterns and maximizes classification accuracy.

For Figure 4, the function selects each chromosome piece linked to a parameter and simplified to include  $i$ -th one bit as part of the modified classification patterns, or converse, if the bit is 0. According to their classification accuracy, every resultant characteristic group needs to be assessed on a set of test data [34]. The fitness value depends on the performance measurement of categorization, like categorization accuracy. Please note that in this instance of

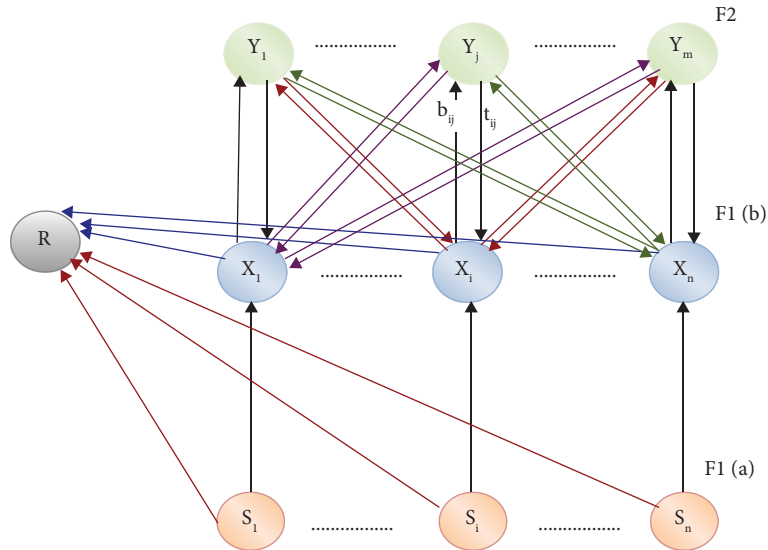


FIGURE 5: Overall architecture of proposed ARCEN network.

selection of features, the binary chromosomes do not contain actual value control factors: information about the existence or lack of features in the optimum classification set is incorporated in the bits themselves for avoiding the need for decoding.

Feature weighting determines the optimum collection of feature weights for the best chromosomes chosen via function selection techniques. In the suggested approach, each gene in the chromosome is given weight. Each weight is regarded as its associated categorization characteristic [37]. This implies that each characteristic is multiplied by its weight before categorization. The weights are limited to a specific interval. The weights of the essential characteristics tend to approach the maximum weight throughout the development. The opposite is true for the less significant characteristics, which behave more like noise and provide modest quantities of discriminatory categorization information. By multiplying features with optimum weights, the feature space is changed to optimize distances between distinct classes in the modified space. The optimum weights of the characteristics are somewhat necessary.

**3.1.4. Classification Using Adaptive Resonance Classifier Network.** The initial neural network classification construction is ARCEN, intended to cluster dual vectors via uncontrolled knowledge. This system controls the extent to identifying patterns that are similar inside one cluster [38]. The network may find clusters based on the data input. The network itself will store the information about clustering the patterns or features without prior knowledge of the potential cluster number and type. As soon as the first input pattern is produced, the network basically “follows the leader” to the next. The second group is created when the space of the first two clusters crosses a predetermined.

It creates a second cluster if the distance between the first and second clusters exceeds a precomputed threshold; otherwise, the pattern is categorized to the primary

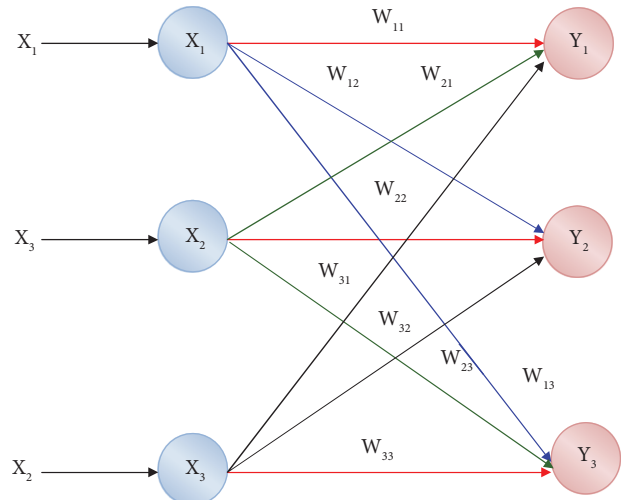


FIGURE 6: Classification network architecture existing in the literature.

cluster, and a similar process is carried out for entire data sets.

**(1) ARCEN Network Architecture.** As shown in Figure 5 of the ARCEN net architecture [39], there are two cluster units and reset units used to check the degree of similarity between the patterns on the unit of a sole cluster. With the aid of two weighted route sets, the F1 layer is connected with F2.

Figure 5 represents the overall framework of a proposed classifier that consists of S input networks, x neurons, and y output layers. The classification process initiated and ends with the various neuron layers interconnected.

**3.2. Comparison with Existing Algorithm.** The computing unit architectures for ARCEN comprise F1 components (participation and boundary components), F2 units (cluster units), and rearrange units that regulate users’ patterns in a

TABLE 4: Analysis of performance metrics with proposed technique.

S. No.	Input signals	Proposed technique			Input measure	Frequency (sec)	Overall accuracy (%)
		Precision	Sensitivity	Specificity			
1	268	0.82	0.864	0.956	0.874	14	91.2
2	389	0.89	0.893	0.978	0.896	19	92.4
3	457	0.90	0.912	0.956	0.945	25	93.9
4	562	0.91	0.923	0.936	0.969	21	97.6

similar cluster degree. Each unit in layer F1(a) is linked to the appropriate unit in layer F1(b). Each unit in F1(a) and F1(b) has its reset unit linked to each unit of F2. Each unit in the F1(b) layer is linked by two weighted routes, each unit in the F2 (cluster) layer. A lower-up weight  $b_{ij}$  links the F1(b) unit  $X_i$  to the F2 unit  $Y_j$ . Similarly, unit  $Y_j$  is linked by top-down  $t_{ji}$  weights to unit  $X_i$ . F2 is the competitive layer with the nonzero activation of the uninhibited node with the most significant net input.

The LVQ network architecture is shown in Figure 6 [40]. It is analogous to a Kohonen map creation with no topological framework to compare the construction of an LVQ network. Every output set is allocated to a specific set, which may be found here.

The existing networks and proposed networks have the major difference in the classification process. The weight and the network layer are different as compared to the existing layer as shown in Figure 6 and Table 3 represents the performance metrics of the proposed work that is calculated from True Positive and True Negative during the classification process.

Table 4 lists accuracy, sensitivity, specificity, F-measure, time, and classification. The average sensitivity for 500 samples is 91 percent, specificity is 98.4 percent, and classification is 92 percent. Precision using LVQ network is seen.

**3.3. Resource Allocation with Proposed Algorithm.** It is difficult to identify the many factors associated with NN, and determining the optimal configuration is time-consuming and requires patience. In BPNN, the MRAN method is employed to determine the smallest number of neurons needed to get included in the layer hidden to achieve maximum efficiency, rather than selecting the design at random or via trial and error. The multilayer perception (MLP) architecture is the most popular neural network architecture, and it is taught through backpropagation (BP) algorithms. The gradient descent technique has been proposed to decrease the mean network squared error as much as possible.

**3.3.1. Networks Present in ARCN.** This model of MRAN incorporates the Resource Allocating Network (RAN) growth criteria of Platt with a pruning approach based on the relative contribution of each remote unit to the overall network output to create a sequential radial learning basis neural network that performs well in both training and testing. The resulting network results in a RAN design that is as simple as possible. Owing to the topological structure and

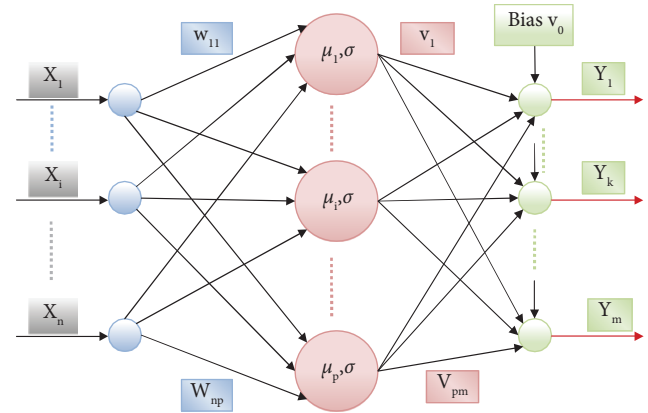


FIGURE 7: Architecture of proposed layers.

openly revealing capability of the learning process, the RBFNN has been particularly compatible for pattern recognition and approximation of function [41, 42]. During the Radial Base Function (RBF) usage, the Gaussian functions were chosen from the essential functions, and the number of units (the Gaussian function widths and centers) can be determined depending on the input features. The output and remote unit weights are determined using a linear procedure described in detail. This method is not well adapted for sequential instruction is the only drawback and often results in an excessive number of hidden units. Platt contributed to addressing these constraints by creating an algorithm that, based on the new input, adds hidden units to the network. As a sequential learning technique, the approach is founded on the principle that the number of units contained in the data must be commensurate with the complexity of the underlying function. This is referred to as Resource Allocating Network (RAN), which begins with 0 hidden units and develops as more units of hidden are allocated, relying upon the novelty of consecutive interpretation. If an observation is made for the first time, the existing network parameters are adjusted using the LMS algorithm to fit the new observation.

**3.3.2. Layer 1 Resource Allocation.** The RAN structure is similar to RBF [43], as depicted in Figure 7. There are two characteristics associated with each hidden unit inside the network, which are referred to as a center ( $r$ ) and a width ( $T$ )

Radially symmetrical operation functions of the hidden units exist in the input space of the hidden units. Only the radial distance between the input vector and the corresponding hidden unit's center parameter will be dependent on the output of each hidden unit. Between the layer hidden

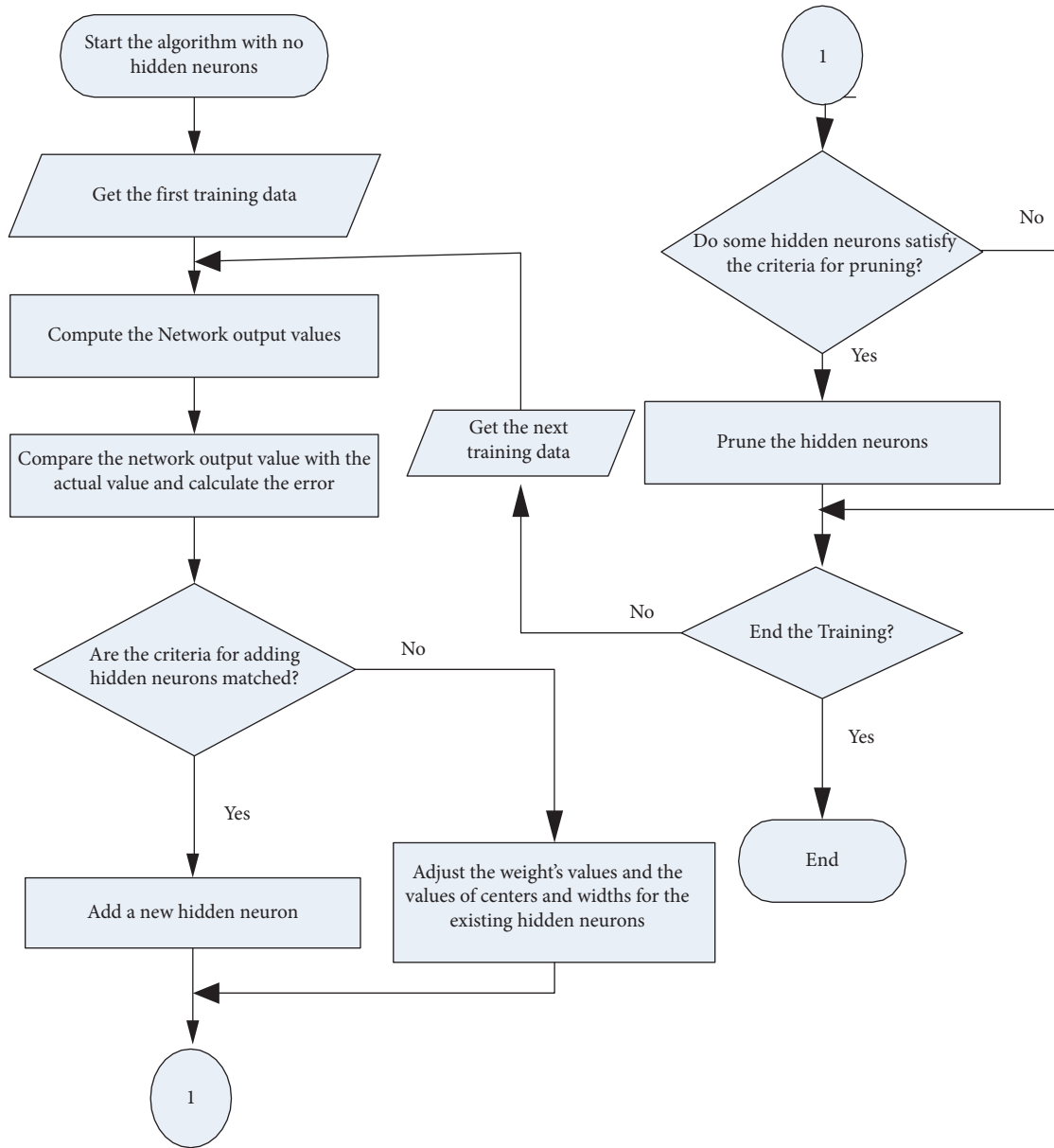


FIGURE 8: MRAN algorithm flowchart.

and input, the weight ranges from  $w_{11}$  to  $w_{np}$ , and between the layer hidden and the output, the weight ranges from  $v_{11}$  to  $v_{pm}$ . The response for each hidden unit is scaled to produce the total output network, which is accomplished via the connecting weights between the units hidden and the output.

Satisfying both aforementioned conditions, the data has been depicted as new, and a new unit hidden is placed into the data structure. Initially, the input is to be located at a distance from the other centers, and second, the error between the output of the network and the estimated output is significant in comparison to the goal output. The minimum precision needed for network output approximation is represented by the  $e_{min}$  value, while the distance  $e_n$  reflects the input resolution scale represented by the distance  $e_n$ . The algorithm begins with the expression  $e_n = e_{max}$ .  $e_{max}$  is

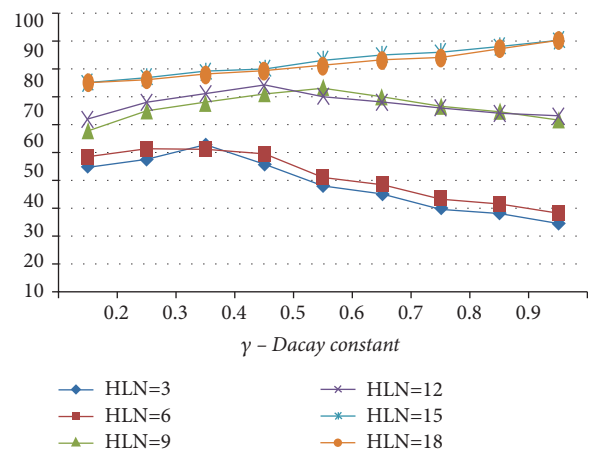


FIGURE 9: MRAN algorithm-based classification for differing decay constant.

TABLE 5: Proposed algorithm parameters.

Trials	No. of neurons in the hidden layer	Amount of test EGG classified (50 each)							Total	Correct classification (%)
		Normal	Abnormality					Bradygastric		
			Dyspepsia	Tachygastric	Nausea	Vomiting	Ulcer			
1 <sup>st</sup>	9	44	39	43	46	34	27	48	372	77.43
2 <sup>nd</sup>	22	43	37	43	39	48	39	49	398	86.24
3 <sup>rd</sup>	23	36	33	33	30	43	47	43	333	63.72
4 <sup>th</sup>	24	46	40	39	43	48	46	44	303	86.67
5 <sup>th</sup>	<b>26</b>	<b>46</b>	<b>42</b>	<b>48</b>	<b>44</b>	<b>49</b>	<b>38</b>	<b>60</b>	<b>326</b>	<b>95.00</b>
6 <sup>th</sup>	26	43	40	43	46	60	44	60	326	90.00
7 <sup>th</sup>	27	43	43	44	46	48	44	49	326	90.00

chosen as the interest scale with the greatest range in the input space, which is often the whole nonzero probability input field. The distance has decreased as  $en = \max e_{\max}, n, e_{\min}$ , where 0 indicates a steady decay and 1 represents a rapid decay. The value of  $en$  is decreased to the value of  $e_{\min}$ . Based on a growing number of observations, the distance criterion may be fine-tuned by using fewer base functions with broad widths (smoother base functions) initially, and lesser width basic functions are assigned to correct the approximation as the number of observations increases. “k” denotes the overlapping factor defining the overlay in the hidden unit input space replies. The value of  $e_{\max}$  and  $e_{\min}$  is 0.4 and 0.2.

Figure 8 shows the flowchart of the MRAN algorithm. It provides details on the calculation of output values of the network through comparison with the real values to get the error value. If the error meets the additional requirement of newly created neurons, then the existing neurons’ weight, center, and breadth are modified to meet the conditions using the new hidden neuron. When the requirements for tailing are met by the neurons hidden, for tailoring the hidden neurons, the training takes place if it does not finish.

(1) *Minimal Resource Allocation Network (MRAN) Training.* The MRAN network is trained by the EGG database. The target is given to 1 for the right position and 0 for the incorrect categorization position. The minimal number of hidden layer neurons (HLN) needed for optimal efficiency is determined using the MRAN method.

Using various decay constant values related to hidden neurons in the layer, Figure 9 depicts the MRAN network consistency in the classification of illness for various decay constants values in the layer [44]. When training MRAN with values lesser compared to 9 HLN, it has been observed that the percentage of classification is lower than expected. For a decay constant of 0.9, 90 percent of the classification is saturated at the 15-point level and above. It has also been noticed that the classification increases linearly with a distinct decreasing constant value for each of the 15 HLN numbers tested. The number of HLN used ranges from 9 to 18 to configure HLN in the BPNN architecture to obtain the highest possible categorization percentage.

Table 5 lists the various research in which MRAN has provided unique neurons, as well as the effects associated with those neurons. The MRAN technique is employed to determine the number of neurons that are required in the

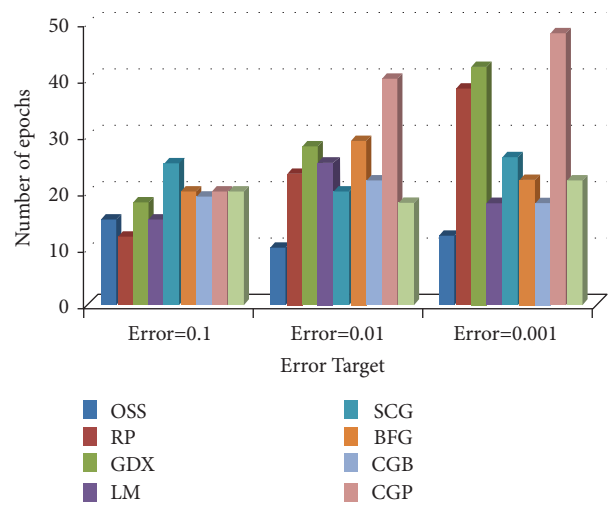


FIGURE 10: Comparison chart of the number of epochs for normal subjects concerning error targets.

layer hidden to achieve maximum efficiency. In the network, 350 samples are being analyzed. Table 4 shows that the 15-neuron network performs at its highest possible efficiency level. Furthermore, the amount of correctly classified [45] for each disease is shown.

Thus, the network [46] set-up using the MRAN algorithm is fixed at 60-15-7. This design is then utilized to classify digestive diseases. Classification is applied by several training algorithms such as *trainidx*, *trainrp*, *trainross*, *trainlm*, *trainscg*, *traincgp*, *trainbfg*, *traincgf*, and *traincgb*. Using accurate classification, the comparison of the training algorithm has been performed.

(2) *Comparison of Training Algorithm Performance in BP-MRAN Network.* Figures 10–16 are presented in the performance comparisons of various training algorithms for normal individuals, subjects of *bradygastric*, subjects of dyspepsia, subjects of nausea, subjects of tachygastric and of ulcers, and subjects of vomiting. The diagram is drawn between the error objective and the periods. From all graphs, the epoch values of *trainrp*, *trainidx*, and *traincgp* have been increasing for different error target values [47].

Figure 10 presents the performance comparisons of various training algorithms for normal individuals and subjects of *bradygastric*.

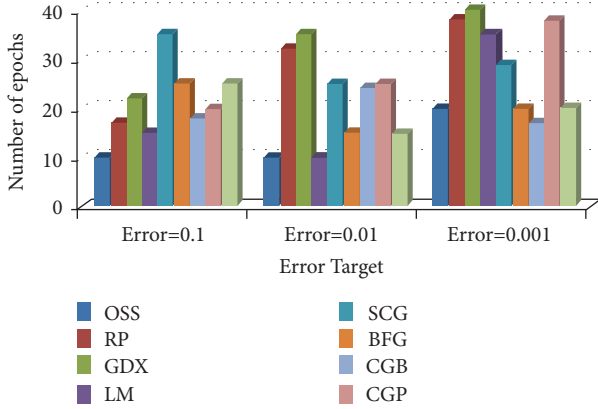


FIGURE 11: Comparison chart of the number of epochs for *bradygastria* subjects concerning error target.

TABLE 6: EGG classification accuracy for various training algorithms with several values of  $\alpha$  and  $\beta$ .

Lr	M	EGG correct classification (%)								
†	‡	OSS	RP	GDX	LM*	SCG	BFG*	CGB	CGP	CGF
( $\alpha$ )	( $\beta$ )									
0.2	0.2	48	81	82	50	54	50	47	82	44
	0.4	52	83	84	52	56	52	52	84	46
	0.6	58	84	88	51	60	58	58	88	48
	0.8	<b>61</b>	86	90	53	58	61	59	90	51
0.4	0.2	45	85	92	52	61	48	45	92	45
	0.4	48	87	<b>94</b>	54	63	48	48	94	48
	0.6	52	91	93	<b>56</b>	62	52	52	93	52
	0.8	58	95	92	55	<b>65</b>	60	58	92	58
0.6	0.2	50	93	91	53	61	52	54	91	50
	0.4	52	95	93	51	63	58	58	93	52
	0.6	58	<b>97</b>	92	54	62	60	<b>62</b>	92	58
	0.8	60	96	90	52	64	<b>63</b>	61	90	<b>60</b>
0.8	0.2	52	90	90	52	60	52	52	90	52
	0.4	54	91	93	53	62	54	54	<b>93</b>	54
	0.6	59	92	92	55	63	56	59	92	59
	0.8	60	95	91	54	62	58	60	91	57

†Lr: rate of learning, ‡ M: momentum factor, \*-M is not applicable.

Figures 11 presents the performance comparisons of various training algorithms for normal individuals.

The categorization of EGG topics using this BP-MRAN architecture takes place with a variable rate of learning and the value of momentum factor for various algorithms of training. Table 5 shows the highest efficiency of every method for every rate of learning and dynamic factor.

The categorization of EGG patients [48] with trainrp is seen in Table 6 at 97% for LR and MF 0.6. Compared to other training algorithms, the trainrp has superior performance. Also, it has been observed that a maximum classification of 61% was recorded with 0.2 Lr and 0.8M for Trainoss. A maximum classification of 94% for Lr and 0.4M was obtained from traingdx algorithm, whereas trainlm algorithm yields a maximum 56% classification for the Lr value 0.4 and 0.6. Meanwhile, the trainscg algorithm yields a maximum 65% classification with 0.4 Lr and 0.8M. The trainbfg algorithm was rated at a maximum level of 63% with 0.6 Lr

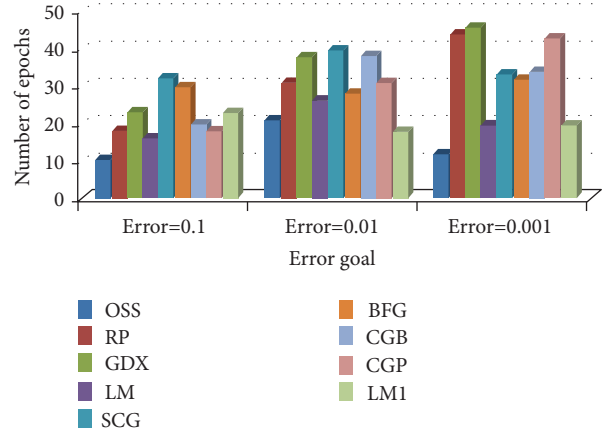


FIGURE 12: Comparison chart of the number of epochs for *dyspepsia* subjects concerning error target.

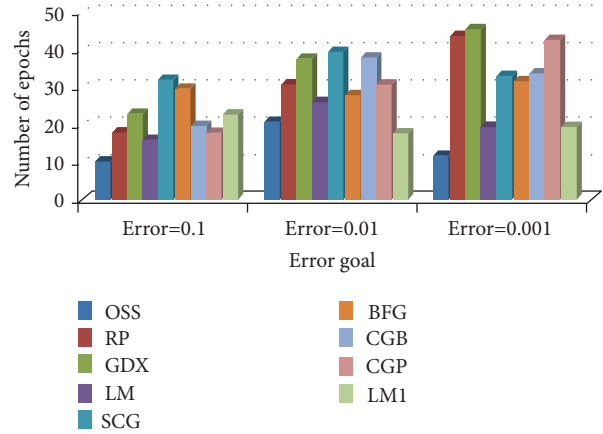


FIGURE 13: Comparison chart of the number of epochs for *nausea* subjects concerning error target.

and 0.8M, the traingcb training algorithm attained a maximum 62% rating for 0.6 Lr and M, the traingcp training algorithm [46] was rated at a maximum 93% level, and the traingcf training algorithm was rated at 60% maximum level.

MRAN, which is employed with the combination of BPNN, determines the total amount of HLN for obtaining optimum classification efficiency [49]. Considering the data of 500 samples, results are calculated following the prior comparative confusion research matrix [50].

#### 4. Confusion Matrix for BP-MRAN Network

For various sample sets, a confusion matrix [51] for signals collected in the laboratory setting of varied compositions was used.

#### 5. Neural Comparison Architectures

This research work will describe the results for the three designs utilized to identify anomalies in the EGG signals. The specificity and accuracy of classification for three different designs are compared [52]. The sensitivity and





FIGURE 14: Comparison chart of the number of epochs for tachygastric subjects concerning error target.

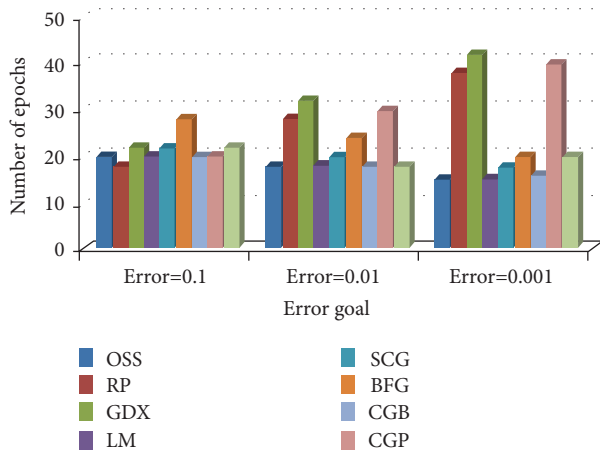


FIGURE 15: Comparison chart of the number of epochs for ulcer subjects concerning error target.

specificities indicate that all architectures are described in Figures 15 and 16. As demonstrated in Figure 16, the classification precision for the three designs is compared. The result shows that a 69.5% accuracy of classification [53] was obtained for ARCNN design, 92.0% was obtained for LVQNN, and 96% was obtained for BP-MRAN.

## 6. Results and Discussion

In this work, three artificial neural network (ANN) architectures have been built and tested to categorize EGG signals. The EGG is classified as normal or abnormal using the ARCNN network, an unsupervised network. The LVQ network is investigated as a supervised method that employs competing layers to improve the accuracy of the classification decision-making process. The BPNN was implemented via the use of supervised learning. Using the MRAN technique, optimizing the architecture's efficiency while simultaneously reducing computation time is possible. The performance of the BP-MRAN training algorithm is compared to that of nine other training algorithms. The most significant percentage of trainrp, traingdx, and traingcp that can be classified is 96.28 percent, 94 percent, and 92.57

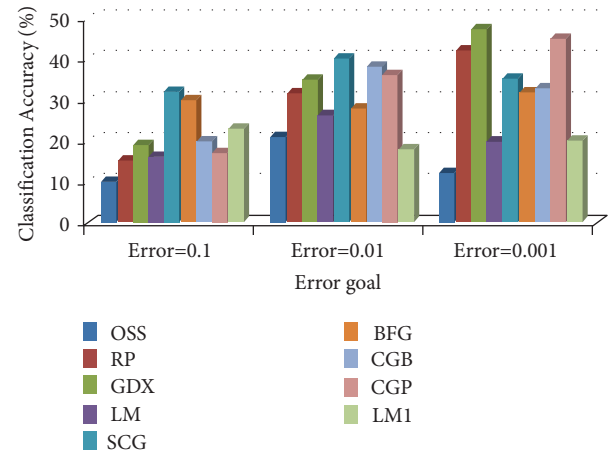


FIGURE 16: Comparison chart of the number of epochs for vomiting subjects concerning error target.

percent, respectively. To compare the sensitivity of different methods, an analysis of specification is performed, and 96% accuracy of classification occurred using BP-MRAN with the combination of a robust algorithm of backpropagation. It was found that the BP-MRAN with trainrp improved performance by 14 percent and 10 percent, respectively, in comparison to the results obtained by Chacon et al. using the BPANN in a combination of trainrp [54] and the results obtained by Curilem et al. using the GA and SVM [55].

## 7. Conclusion

Stomach disorders bring significant changes in the physical health system, which also affects the mental health of humans. The proposed work uses various machine learning algorithms to classify stomach disorders. The primary goal was to categorize electrogastrigraphy data into two or three classes accurately. The electrogastrigraphy signals are classified into three classes (normogastria, bradygastric, and tachygastric) and two classes. In future work, various deep learning algorithms can be carried out with the proposed technique, which would give better results while using many datasets.

## Data Availability

The data that support the findings of this study are available on request from the corresponding author.

## Conflicts of Interest

The authors declare that they have no conflicts of interest to report regarding the present study.

## References

- [1] B. O. Familoni, Y. J. Kingma, and K. L. Bowes, "Noninvasive assessment of human gastric motor function," *IEEE Transactions on Biomedical Engineering*, vol. 34, no. 1, pp. 30–36, 1987.

- [2] W. Sha, P. J. Pasricha, and J. D. Z. Chen, "Rhythmic and spatial abnormalities of gastric slow waves in patients with functional dyspepsia," *Journal of Clinical Gastroenterology*, vol. 43, no. 2, pp. 123–129, 2009.
- [3] C. L. Chen, H. H. Lin, W. C. Orr, C. C. H. Yang, and T. B. J. Kuo, "Transfer function analysis of heart rate variability in response to water intake: correlation with gastric myoelectrical activity," *Journal of Applied Physiology*, vol. 96, no. 6, pp. 2226–2230, 2004.
- [4] H. Geldof, E. J. van der Schee, A. J. Smout, J. P. van de Merwe, M. van Blankenstein, and J. L. Grashuis, "The myoelectrical activity of the stomach in gastric ulcer patients: an electrogastrographic study," *Neuro-Gastroenterology and Motility*, vol. 1, no. 2, pp. 122–130, 1989.
- [5] M. P. Mintchev, Y. J. Kingma, and K. L. Bowes, "Accuracy of cutaneous recordings of gastric electrical activity," *Gastroenterology*, vol. 104, no. 5, pp. 1273–1280, 1993.
- [6] A. J. P. M. Smout, E. J. Van der Schee, and J. L. Grashuis, "What is measured in electrogastrography?" *Digestive Diseases and Sciences*, vol. 25, no. 3, pp. 179–187, 1980.
- [7] J. Chen, J. Vandewalle, W. Sansen, G. Vantrappen, and J. Janssens, "Adaptive method for cancellation of respiratory artefact in electrogastric measurements," *Medical, & Biological Engineering & Computing*, vol. 27, no. 1, pp. 57–63, 1989.
- [8] J. K. Stevens and N. Worrall, "External recording of gastric activity: the electrogastrogram," *Physiological Psychology*, vol. 2, no. 2, pp. 175–180, 1974.
- [9] J. D. Z. Chen, W. R. Stewart, and R. W. McCallum, "Spectral analysis of episodic rhythmic variations in the cutaneous electrogastrogram," *IEEE Transactions on Biomedical Engineering*, vol. 40, no. 2, pp. 128–135, 1993.
- [10] H. P. Simonian, K. Panganamamula, H. P. Parkman et al., "Multichannel electrogastrography (EGG) in normal subjects: a multicenter study," *Digestive Diseases and Sciences*, vol. 49, no. 4, pp. 594–601, 2004.
- [11] W. Ding, S. Qin, L. Miao, N. Xi, H. Li, and Y. Wang, "Processing and analysis of bio-signals from human stomach," in *Proceedings of the 2010 IEEE International Conference on Robotics and Biomimetics*, pp. 769–772, IEEE, Tianjin, China, December 2010.
- [12] L. Zhenghu, Y. Jingan, and S. Chenggan, "A new method for processing EGG signals based on wavelet transform," vol. 1, pp. 353–358, in *Proceedings of the WCC 2000-ICSP 2000. 2000 Fifth International Conference on Signal Processing Proceedings. 16th World Computer Congress*, vol. 1, IEEE, Beijing, China, August 2000.
- [13] M. Kania, M. Fereniec, and R. Maniewski, "Wavelet denoising for multi-lead high resolution ECG signals," *Measurement Science Review*, vol. 7, no. 4, pp. 30–33, 2007.
- [14] C. Zhongjia, L. Xiaohu, L. Yanjie, and C. Cheng, "The research of affecting factors of bio-fuel high dense shaping at room temperature based on SPSS," vol. 2, pp. 762–765, in *Proceedings of the 2010 International Conference on Digital Manufacturing & Automation*, vol. 2, IEEE, Changsha, China, December 2010.
- [15] W. Xing-ce, X. Feng, L. Chang, Z. Ming-quan, W. Zhong-ke, and L. Xin-yu, "The study of pre-processing method of brain vessel segmentation based on parameterized statistical model," in *Proceedings of the 2010 IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA)*, pp. 90–94, IEEE, Changsha, China, September 2010.
- [16] A. M. Elsayad, "Classification of ECG arrhythmia using learning vector quantization neural networks," in *Proceedings of the 2009 International Conference on Computer Engineering & Systems*, pp. 139–144, IEEE, Cairo, Egypt, December 2009.
- [17] A. Kandaswamy, C. S. Kumar, R. P. Ramanathan, S. Jayaraman, and N. Malmurugan, "Neural classification of lung sounds using wavelet coefficients," *Computers in Biology and Medicine*, vol. 34, no. 6, pp. 523–537, 2004.
- [18] T. N. Tsung-Nan Lin, C. L. Giles, B. G. Horne, and S. Y. Sun-Yuan Kung, "A delay damage model selection algorithm for NARX neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2719–2730, 1997.
- [19] S. Dutta, A. Chatterjee, and S. Munshi, "Identification of ECG beats from cross-spectrum information aided learning vector quantization," *Measurement*, vol. 44, no. 10, pp. 2020–2027, 2011.
- [20] M. R. Ahsan, M. I. Ibrahimy, and O. O. Khalifa, "Neural network classifier for hand motion detection from EMG signal," in *Proceedings of the Fifth Kuala Lumpur International Conference on Biomedical Engineering 2011*, pp. 536–541, Springer, Kuala Lumpur, Malaysia, June 2011.
- [21] Y. C. Shiao, H. T. Chian, J. L. Su, S. T. Fan, and T. S. Kuo, "Using neural network for cardiac motion analysis of Tc-99m labeled RBC radionuclide angiography," in *Proceedings of the 2011 Fifth International Conference on Bioinformatics and Biomedical Engineering*, pp. 1–3, IEEE, Wuhan, China, May 2011.
- [22] A. Barrea and R. Barrea, "Local fuzzy c-means clustering for medical spectroscopy images," *Applied Mathematical Sciences*, vol. 5, no. 30, pp. 1449–1458, 2011.
- [23] J. Chen and R. W. McCallum, "Electrogastrography: measurement, analysis and prospective applications," *Medical, & Biological Engineering & Computing*, vol. 29, no. 4, pp. 339–350, 1991.
- [24] J. W. Hamilton, B. E. Bellahsene, M. Reichelderfer, J. G. Webster, and P. Bass, "Human electrogastrograms," *Digestive Diseases and Sciences*, vol. 31, no. 1, pp. 33–39, 1986.
- [25] B. Krusiec-Swidergoł and K. Jonderko, "Multichannel electrogastrography under a magnifying glass—an in-depth study on reproducibility of fed state electrogastrograms," *Neuro-Gastroenterology and Motility*, vol. 20, no. 6, pp. 625–634, 2008.
- [26] F. Dirgenali, S. Kara, and S. Okkesim, "Estimation of wavelet and short-time Fourier transform sonograms of normal and diabetic subjects' electrogastrogram," *Computers in Biology and Medicine*, vol. 36, no. 12, pp. 1289–1302, 2006.
- [27] B. Pfaffenbach, R. J. Adamek, K. Kuhn, and M. Wegener, "Electrogastrography in healthy subjects," *Digestive Diseases and Sciences*, vol. 40, no. 7, pp. 1445–1450, 1995.
- [28] H. P. Parkman, A. D. Harris, M. A. Miller, and R. S. Fisher, "Influence of age, gender, and menstrual cycle on the normal electrogastrogram," *American Journal of Gastroenterology*, vol. 91, no. 1, pp. 127–33, 1996.
- [29] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *Journal of Fourier Analysis and Applications*, vol. 4, no. 3, pp. 247–269, 1998.
- [30] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, "Textural features for image classification," *IEEE Transactions on systems, man, and cybernetics*, vol. 3, no. 6, pp. 610–621, 1973.
- [31] T. Chang and C.-C. J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Transactions on Image Processing*, vol. 2, no. 4, pp. 429–441, 1993.
- [32] W. Siedlecki and J. Sklansky, "A note on genetic algorithms for large-scale feature selection," *Handbook of Pattern Recognition and Computer Vision*, vol. 10, pp. 88–107, 1993.

- [33] B. Bhanu and Y. Lin, "Genetic algorithm based feature selection for target detection in SAR images," *Image and Vision Computing*, vol. 21, no. 7, pp. 591–608, 2003.
- [34] D. P. Muni, N. R. Pal, and J. Das, "Genetic programming for simultaneous feature selection and classifier design," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 36, no. 1, pp. 106–117, 2006.
- [35] E. Zio, P. Baraldi, and N. Pedroni, "Selecting features for nuclear transients classification by means of genetic algorithms," *IEEE Transactions on Nuclear Science*, vol. 53, no. 3, pp. 1479–1493, 2006.
- [36] C.-L. Huang and C.-J. Wang, "A GA-based feature selection and parameters optimization for support vector machines," *Expert Systems with Applications*, vol. 31, no. 2, pp. 231–240, 2006.
- [37] M. L. Raymer, W. F. Punch, E. D. Goodman, L. A. Kuhn, and A. K. Jain, "Dimensionality reduction using genetic algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 4, no. 2, pp. 164–171, 2000.
- [38] R. Feraund, O. J. Bernier, J.-E. Viallet, and M. Collobert, "A fast and accurate face detector based on neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 1, pp. 42–53, 2001.
- [39] K. Umamaheswari, S. Sumathi, S. N. Sivanandam, and K. K. N. Anburajan, "Efficient finger print image classification and recognition using neural network data mining," in *Proceedings of the 2007 International Conference on Signal Processing, Communications and Networking*, pp. 426–432, IEEE, Chennai, India, February 2007.
- [40] A. Motwani, P. K. Shukla, and M. Pawar, "Novel framework based on deep learning and cloud analytics for smart patient monitoring and recommendation (SPMR)." *Journal of Ambient Intelligence and Humanized Computing*, 2021.
- [41] M. T. Musavi, W. Ahmed, K. H. Chan, K. B. Faris, and D. M. Hummels, "On the training of radial basis function classifiers," *Neural Networks*, vol. 5, no. 4, pp. 595–603, 1992.
- [42] K. M. Tao, "A closer look at the radial basis function (RBF) networks," in *Proceedings of the Twentyseventh Asilomar Conference on Signals, Systems and Computers*, pp. 401–405, IEEE, Pacific Grove, CA, USA, November 1993.
- [43] L. Lu Yingwei, N. Sundararajan, and P. Saratchandran, "Performance evaluation of a sequential minimal radial basis function (RBF) neural network learning algorithm," *IEEE Transactions on Neural Networks*, vol. 9, no. 2, pp. 308–318, 1998.
- [44] V. Roy, P. K. Shukla, A. K. Gupta, V. Goel, P. K. Shukla, and S. Shukla, "Taxonomy on EEG artifacts removal methods, issues, and healthcare applications," *Journal of Organizational and End User Computing*, vol. 33, no. 1, pp. 19–46, 2021.
- [45] P. K. Shukla, J. Kaur Sandhu, A. Ahirwar, D. Ghai, P. Maheshwary, and P. K. Shukla, "Multiobjective genetic algorithm and convolutional neural network based COVID-19 identification in chest X-ray images," *Mathematical Problems in Engineering*, vol. 2021, Article ID 7804540, 9 pages, 2021.
- [46] A. Sedik, M. Hammad, F. E. Abd El-Samie, B. B. Gupta, and A. A. Abd El-Latif, "Efficient Deep Learning Approach for Augmented Detection of Coronavirus Disease," *Neural Comput & Applic*, vol. 19, 2021.
- [47] S. Pandit, P. K. Shukla, A. Tiwari, P. K. Shukla, and R. Dubey, "Review of video compression techniques based on fractal transform function and swarm intelligence," *International Journal of Modern Physics B*, vol. 34, no. 8, Article ID 2050061, 2020.
- [48] G. Khambra and P. Shukla, "Novel machine learning applications on fly ash based concrete: an overview," *Materials Today Proceedings*, pp. 2214–7853. In press, 2021.
- [49] M. Hammad, M. H. Alkinani, B. B. Gupta, and A. A. Abd El-Latif, "Myocardial Infarction Detection Based on Deep Neural Network on Imbalanced Data," *Multimedia Systems*, 2021.
- [50] M. Hammad, A. M. Iliyasu, A. Subasi, E. S. L. Ho, and A. A. A. El-Latif, "A multitier deep learning model for arrhythmia detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, Article ID 2502809, 2021.
- [51] A. S. Alghamdi, K. Polat, A. Alghoson, A. A. Alshdadi, and A. Ahmed, "Gaussian process regression (GPR) based non-invasive continuous blood pressure prediction method from cuff oscillometric signals," *Applied Acoustics*, vol. 164, p. 2020, 107256, <https://doi.org/10.1016/j.apacoust.2020.107256>.
- [52] A. S. Alghamdi, K. Polat, A. Alghoson, A. A. Alshdadi, A. Ahmed, and Abd El-Latif, "A novel blood pressure estimation method based on the classification of oscillometric waveforms using machine-learning methods," *Applied Acoustics*, vol. 164, Article ID 107279, 2020.
- [53] B. Abd-El-Atty, A. M. Iliyasu, H. Alaskar, and A. A. Abd El-Latif, "A robust quasi-quantum walks-based steganography protocol for secure transmission of images on cloud-based E-healthcare platforms," *Sensors*, vol. 20, no. 11, p. 3108, 2020.
- [54] G. Curilem, J. Vergara, G. Fuentealba, G. Acuña, and M. Chacón, "Classification of seismic signals at Villarrica volcano (Chile) using neural networks and genetic algorithms," *Journal of Volcanology and Geothermal Research*, vol. 180, no. 1, pp. 1–8, 2009.
- [55] M. Curilem, M. Chacón, G. Acuña et al., "Comparison of artificial neural networks a support vector machines for feature selection in electrogastrography signal processing," in *Proceedings of the 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pp. 2774–2777, IEEE, Buenos Aires, Argentina, September 2010.

## Research Article

# An Integrated Approach for Cancer Survival Prediction Using Data Mining Techniques

Ishleen Kaur <sup>1</sup>, M. N. Doja,<sup>1</sup> Tanvir Ahmad <sup>1</sup>, Musheer Ahmad <sup>1</sup>, Amir Hussain <sup>2</sup>,  
Ahmed Nadeem <sup>3</sup>, and Ahmed A. Abd El-Latif <sup>4</sup>

<sup>1</sup>Department of Computer Engineering, Jamia Millia Islamia, New Delhi 110025, India

<sup>2</sup>School of Computing, Edinburgh Napier University, Merchiston Campus, Edinburgh, Scotland EH10 5DT, UK

<sup>3</sup>Department of Pharmacology & Toxicology, College of Pharmacy, King Saud University,  
PO Box 2455, Riyadh 11451, Saudi Arabia

<sup>4</sup>Department of Mathematics and Computer Science, Faculty of Science, Menoufia University, Shibin El Kom 32511, Egypt

Correspondence should be addressed to Ahmed A. Abd El-Latif; [a.rahim@gmail.com](mailto:a.rahim@gmail.com)

Received 4 November 2021; Accepted 27 November 2021; Published 28 December 2021

Academic Editor: Anastasios D. Doulamis

Copyright © 2021 Ishleen Kaur et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Ovarian cancer is the third most common gynecologic cancers worldwide. Advanced ovarian cancer patients bear a significant mortality rate. Survival estimation is essential for clinicians and patients to understand better and tolerate future outcomes. The present study intends to investigate different survival predictors available for cancer prognosis using data mining techniques. Dataset of 140 advanced ovarian cancer patients containing data from different data profiles (clinical, treatment, and overall life quality) has been collected and used to foresee cancer patients' survival. Attributes from each data profile have been processed accordingly. Clinical data has been prepared corresponding to missing values and outliers. Treatment data including varying time periods were created using sequence mining techniques to identify the treatments given to the patients. And lastly, different comorbidities were combined into a single factor by computing Charlson Comorbidity Index for each patient. After appropriate preprocessing, the integrated dataset is classified using appropriate machine learning algorithms. The proposed integrated model approach gave the highest accuracy of 76.4% using ensemble technique with sequential pattern mining including time intervals of 2 months between treatments. Thus, the treatment sequences and, most importantly, life quality attributes significantly contribute to the survival prediction of cancer patients.

## 1. Introduction

Cancer, along with coronary heart diseases, accounts for most deaths globally (the top 10 causes of death [1]). The incidence rate of cancer has increased over the past few decades. It has been estimated that 1 in 9 Indians will develop cancer during their lifetime. According to GLOBOCAN, India recorded the highest number of deaths globally in ovarian cancer. Ovarian cancer is the third most common site of cancer among women in India. It is also the third most commonly occurring gynecologic cancer worldwide and has the worst mortality rate. Clinicians and scientists have been conducting great experiments and research to predict cancer patients' survivability [2, 3]. Yet, there are no quality survival

estimation predictors available. Survival estimation predictors are essential for clinicians to precisely adopt the treatments and medications for the patients.

Data-driven prediction techniques can assist in a better cancer prognosis model. Since its origin, data mining techniques have been efficaciously used in many healthcare research kinds, especially cancer management [4, 5]. The medical models based on data mining techniques can capture intricate details and patterns in data. Several studies involve online datasets like UCI machine learning, SEER [6], and TCGA [7]. However, these datasets only cover datasets from western countries or only from a limited area. Although the number of instances in online datasets is large, these might not capture the region-specific analyses. It has

been proven in past studies that race and region can play a significant role in the survivability of cancer patients [8]. Conversely, clinical studies having fewer instances can capture more local aspects of cancer patients and their management. The present research focuses on various attributes that can be significant predictors in estimating the survival of advanced ovarian carcinoma patients that are mostly unavailable in online datasets.

The existing literature focuses on including different clinical attributes like age, CA-125 levels, histology, and stage to investigate the survivability and mortality of ovarian cancer patients [9]. Some of the researchers also intended to explore the outcome for patients treated with neoadjuvant chemotherapy (NACT) [10] or surgery [11]. Nonetheless, the current literature lacks proper research that may give insights into ovarian cancer survival using machine learning approaches and since its initiation, machine learning technology has progressed a lot and is proven to provide good results in almost every area. Some studies performed statistical analysis to find the correlation of treatments with survival [10]. But most of these studies are a part of clinical trials having a controlled environment. A retrospective study in an uncontrolled setting with a variety of participants can point out some useful insights that might not be possible with a clinical trial dataset. Also, to the best of the authors' knowledge, no existing literature emphasized the different sequences of treatments for ovarian cancer patients. Furthermore, various comorbidities can play an essential role in the overall health of patients [12]. The present study involves recording and using some relevant predictors for survival analysis of cancer patients and clinical attributes. These attributes were not available in any online datasets. The collected and processed features can be used and extended to survive any cancer or other serious condition study.

The present study aims to identify the significance of different predictors for advanced ovarian carcinoma patients. An integrated model using attributes from different data profiles can assist in a robust model for predicting survival outcome of patients. The attributes from different data profiles have been collected from a cancer hospital and processed accordingly. Cancer patients are given multiple lines of treatment to prolong their survival. The present study is an attempt to identify the different lines of treatments given to ovarian cancer patients using sequence mining approaches. These treatments and the estimated time elapsed among treatments might contribute some valuable perceptions to the survival of patients. Previous literature has also acknowledged the association of time between treatments with prognosis in patients with ovarian carcinoma [13]. Life quality attributes like performance status and comorbidities also have a significant impact on any person's survival. These attributes have been explored in the study to examine their effect on survival.

The remainder of the study is structured as follows: Section 2 presents a brief background on ovarian cancer and its prevalence in India. Section 3 provides some of the existing literature on ovarian cancer survival analysis. Section 4 explains in detail the dataset and the proposed methodology in the study. The study's results and discussion

and its comparison to the existing methods are given in Section 5. Section 6 discusses some of the study's limitations, and the conclusion is presented in Section 7.

## 2. Background

*2.1. Ovarian Cancer.* Ovarian cancer has the worst mortality in all gynecologic cancers. Overweight and obese women have a higher risk of ovarian cancer [14]. Age is also a significant factor in cancer incidence. While its incidence rates have remained constant in some European countries, Asia has experienced increased incidence rates from the past few decades [15]. Survival rates are less than 20% in Indian women. According to a report, 50% of India's total ovarian cancer cases occurred at 45–65 years [16]. Though, most of the western countries have a median range of more than 60 [17].

Ovarian cancer can have around 90% survival rates if detected in the early stage. However, reports have shown that most patients are diagnosed in later stages, with survival less than 40% (SEER Program). This is why it is also known as “silent killer” because more than 60% of the cases are diagnosed at advanced stages (Stages III and IV). Epithelial ovarian cancer is the most common, including high-grade serous, low-grade serous, endometrioid, clear cell, and mucinous. Patients diagnosed in most advanced ovarian cancer cases are provided with multiple lines of treatment. These include cytoreductive surgery (CRS) with adjuvant chemotherapy, or neoadjuvant chemotherapy (NACT) with Interval debulking surgery (IDS) and adjuvant chemotherapy, or hormonal therapy or chemotherapy [18].

*2.2. Sequence Mining.* A sequence ‘seq’ is a collection of ordered symbols.  $|seq|$  denotes the length of the sequence [19]. A substring of a sequence is a collection of consecutive symbols of the sequence. However, in a subsequence, the symbols need not be consecutive. For example, if PQRS is a sequence with symbols {P, Q, R, S}, then both PQS and PQR can be subsequences of the sequence. But PQS is not a substring of the mentioned sequence. Sequence mining refers to identifying frequently occurring subsequences from a database of sequences. The user determines the term “frequent” by varying the support of the sequences. Support of 0.5 suggests that the database should contain a subsequence in at least 50% of the sequences.

Researchers have devised several sequence mining algorithms. Generalized Sequential Pattern (GSP) [20] is one of the first sequence mining algorithms formulated on the basis of Apriori algorithm [21]. GSP works by identifying the subsequences by scanning the dataset and computing their support. Subsequences with support less than the threshold support are removed from further analysis. For  $k$  length sequences, GSP scans the dataset  $k$  times. Once the frequent sequence at level  $k$  ( $k$  length sequence) is found, a candidate for length  $k + 1$  is generated. Various other researchers also tried to formulate sequence mining algorithms with less time and space complexity. SPADE and PrefixSpan are examples of such algorithms [22, 23].

### 3. Related Work

Several researchers tried to analyze advanced ovarian cancer patients' survival using statistical and conventional survival methods concerning different survival estimators. Vincent et al. [10] used univariate analysis to identify the prognostic factors for stage 3c or 4a ovarian cancer patients. The dataset was collected from 11 French centers and included 483 patients who were treated with NACT followed by surgery. Univariate analysis showed that the absence of cytoreductive surgery (CRS) was associated with worse survival. Similarly, CA-125 value higher or equal to 3000 U/ml had decreased overall survival.

Deng et al. [24] used data from the online dataset SEER (SEER Program) to analyze the survival based on metastatic site for stage 4 patients. Due to the publicly available online dataset, the number of patients is higher than that in other clinical studies. After various inclusions and exclusions, the data analyzed consisted of 1481 patients. Univariate and multivariate analyses showed that the most common sites of metastasis are liver followed by lymph nodes. For patients with lung metastases only, patients who received chemotherapy had a higher survival than those who did not receive chemotherapy. Surgery was also associated with higher survival rates in patients with lymph nodes and liver metastases, but it was not a significant self-determining aspect in patients having lung metastasis. Akhavan et al. [12] also conducted statistical chi squared and Student's test on a dataset collected from Tehran to investigate the effect of diabetes on ovarian cancer survival. The histology considered in particular was epithelial carcinoma. The results suggested that the patients having diabetes had poor overall and progression free survival than those without diabetes.

In a more recent study [25], the authors collected a dataset of around 460 patients from a cancer center to compare white women's survival with black women. The dataset included 365 white patients and 95 black women. It was observed that more white women received surgery, chemotherapy, or surgery chemotherapy sequence. It was also revealed that despite receiving the same treatment sequence, black women had higher mortality rates from ovarian cancer.

Clinicians and researchers from Indian hospitals also conducted statistical tests for survival analysis of advanced epithelial ovarian carcinoma patients. Viswanathan et al. [26] analyzed the data of stage 3 or stage 4 advanced epithelial carcinoma patients diagnosed in years 2015–2018. 111 patients were analyzed by the authors, of which the majorities were of serous histology. Most of the patients were given NACT followed by CRS. It was observed that CRS had improved overall survival and progression-free survival. Also, patients with optimal CRS after NACT had significantly lower recurrence rates and better survival than those suboptimally cytoreduced.

Tseng et al. [27] tried to identify the risk factors in women with ovarian cancer prominent in terms of cancer recurrence. Data mining techniques were used separately using leave one out cross-validation to rank the factors. Since individual data mining techniques cannot address the

problem efficiently, the authors used an ensemble approach. The ensemble approach obtained better results than the pure classification techniques, with C5.0 achieving 90% accuracy. Various authors have also used machine learning techniques to predict ovarian cancer. Lu et al. [28] used a decision tree model and feature selection measures to predict the occurrence of ovarian cancer using different blood routine tests, chemistry, and tumor markers. Several other studies also used different classification techniques to predict survival in various types of cancer [29, 30]. However, most of the studies involved online datasets confined to only a specific country or area. While some of those results can be generalized to other regions, it is a well-known fact that cancer behaves differently with different environment and socioeconomic status of the patients [8].

### 4. Methodology

The proposed methodology of the study is given in Figure 1. This study's approach is divided into three major steps, including data collection, data preprocessing, and classification. The main essence of the study is involved in the dataset used for the analysis. The proposed approach follows an integrated methodology that uses data from three different profiles. However, the medical dataset suffers from many missing and irrelevant data that cannot be directly used for classification. Hence, the second step of the approach involves preprocessing of the dataset according to their data profiles. While clinical data is prepared using standard imputation techniques, we have employed sequence mining techniques to generate treatment sequences given to the patients. Similarly, attributes measuring life quality are created to capture the overall well-being of patients. After all the preprocessing, classification techniques are applied to the integrated dataset. Each step is explained in detail in the following subsections.

*4.1. Data Collection.* This study is based on a dataset collected from a hospital located in New Delhi, India. The case study used for the analysis is of advanced ovarian cancer. The data was collected from the hospital manually from the files digitally stored in the hospital's repository after obtaining appropriate approval from the hospital's Scientific Committee. The study got a waiver from the IRB of the hospital due to anonymity in the use of data. Due to the ethics policies of the hospital, data cannot be shared publicly. The data collected includes three kinds of attributes-clinical attributes, treatment attributes, and comorbidities data. Clinical characteristics including CA-125 levels at the time of diagnosis, presence of ascites, grade, FIGO substage, and histology were collected and recorded for each patient. CA-125 levels denote a diagnostic attribute for ovarian cancer. The presence of ascites and cancer grade define the overall extent and aggressiveness of cancer cells in the body. Higher CA-125 levels, ascites presence, and grade suggest aggressive cancer. Since the collected dataset included advanced cancer patients only, the majority of the patients had stage III or stage IV cancer. Since we have used FIGO substage, stage III

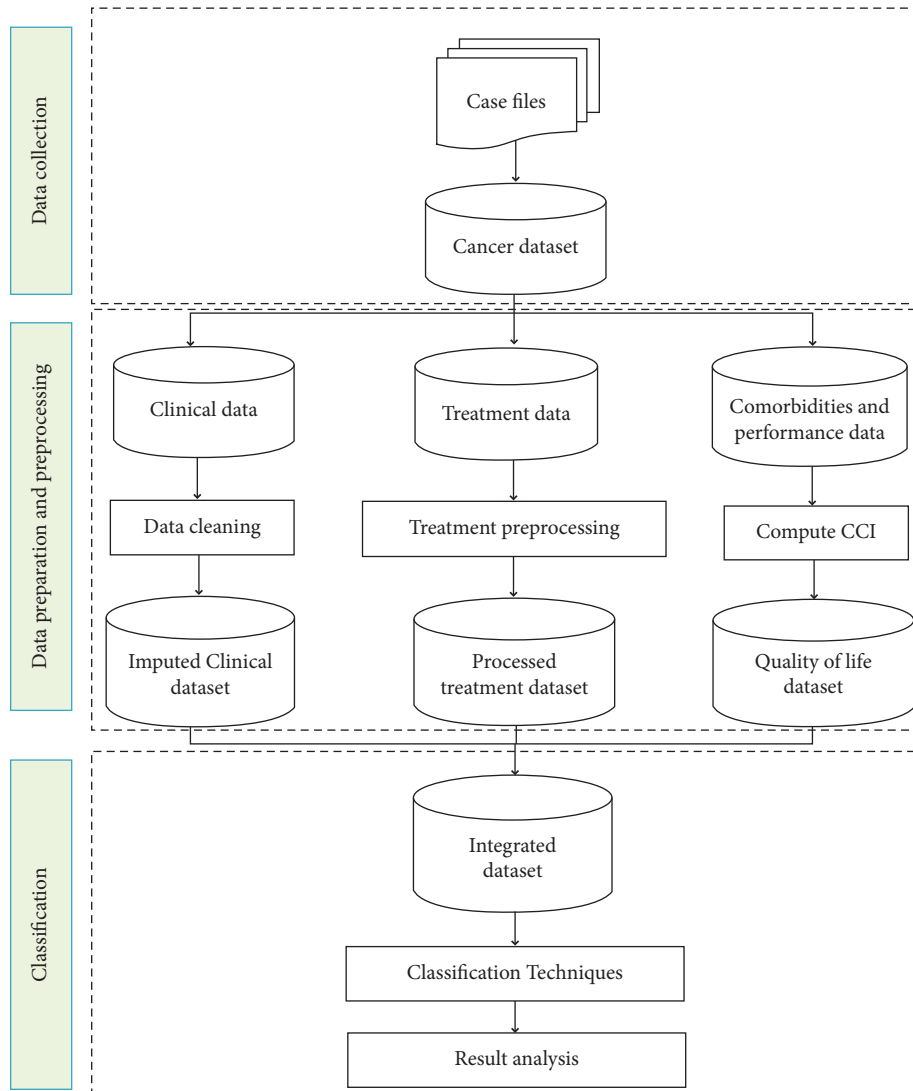


FIGURE 1: Methodology followed in study.

cancer patients were further divided into stages 3a, 3b, and 3c cancer. Clinical data has proven to have a high association with the survival and be the most widely used predictors in the existing studies.

Unlike the online datasets and other clinical studies, the present study also collected treatments and appropriate time intervals between each set of treatments given to each patient. The correct treatments given to the patients can prolong their survival. Also, the time elapsed between these treatments might suggest a better or worse response to the treatments. The treatments and the time intervals thus can aid a better survival model. A total of four lines of treatments were recorded for each patient. Most of the patients received less than four treatment lines for three years.

Also, ECOG levels indicating each patient's performance levels were recorded along with several comorbidities like diabetes, heart disease, and hypertension of each patient. The significance of ECOG levels and comorbidities has also been acknowledged in survival analysis of other cancer types [31]. The inclusion of life quality attributes can suggest the overall

well-being of the patients and thus can better predict the overall survival of the patients. For a better comparison with existing studies and to include recent and relevant data, patients identified in the years after 2011 and before 2015 were used to collect data. The specified time range also allows for the proper retrieval of survival information of 3 years. Survival of 3 years was collected from the hospital's files or by directly contacting the patient or patient's family.

#### 4.2. Data Preprocessing and Analysis

**4.2.1. Data Preparation and Preprocessing.** All the relevant details and information collected in the previous step were recorded and maintained in a spreadsheet. Each attribute category has been handled accordingly to gain a better perspective and improve patients' overall survival prediction.

**Clinical Data Preprocessing.** Clinical data has been cleaned to remove any outliers and handle missing data. Any instance with missing survival information was removed from the

analysis to create a reliable model. Further, instances with more than fifty percent missing data were also removed as larger missing data values can lead to a weak model. The dataset after removing these patients' cases consisted of 149 patients. The rest of the missing data was handled by using mean and mode imputation techniques. Since there were only 9 cases with missing data left and mostly categorical attributes (e.g., presence of ascites), techniques like k-NN imputation did not perform well. Thus, in all the leftover instances, missing numerical attributes were filled out with the mean value of the patients' same class. Similarly, instances with missing nominal attributes were filled with the mode value of the same class. The same has been carried out with MATLAB software using `rmissing()` and `fillmissing()` in-built methods.

*Treatment Data Preprocessing.* The study's objective was to process the data based on each attribute's category and behavior. Treatment preprocessing performed for this study is shown in Figure 2.

The sequences of treatments were processed by creating a database of treatment sequences for each patient. The database generated was supplied to modified sequence mining algorithm GSP. GSP was adjusted to obtain frequent treatment substrings, i.e., treatment sequences with no-gap constraint. The sequence mining algorithm has been implemented in Java [32]. 0.05 value of support has been used for the study to collect the maximum sequences of treatments possible. The no-gap constraint means that intermediate therapies would not be considered for frequent sequences; i.e., if a patient received treatment W in between treatments X and Y, then  $X \rightarrow Y$  is not a valid recurring sequence. The no-gap constraint is attained at the time of counting support of each candidate sequence. The resultant treatment sequences are mentioned as follows.

The time intervals are applied in the resultant frequent treatment sequences, as shown in Figure 3. The time intervals chosen belonged to the 6 months range, i.e.,  $\leq 6$ , 7–12, 13–18, till 31–36. The selection of time intervals was intuitive for 3-year survival and based on the previous literature [33]. Yet, when the data was analyzed, it was observed that most of the patients received their next treatments within 6–8 months of the previous treatment. This may be attributed to the clinical implications of treatments for advanced ovarian cancer patients to commence treatments early [13]. Thus, varying time intervals were chosen (e.g., one month, two months, and three months, till six months) to determine the prognostic value of different time intervals in ovarian cancer survival. A binary matrix is then created based on the attributes. If a patient receives a treatment Y within one month of treatment X, then  $\{X \ T_1 \ Y\}$  column will be marked 1. Since time intervals of 1 month, two months, or three months resulted in 36, 18, or 12 time ranges, it resulted in a wide array of attributes to be applied for classification. Thus, an attribute selection measure was used for the binary matrix. Information gain was computed for each attribute, and attributes having information gain greater than 0 were used for further analysis. The information gain can be calculated using the formula given by the following equations:

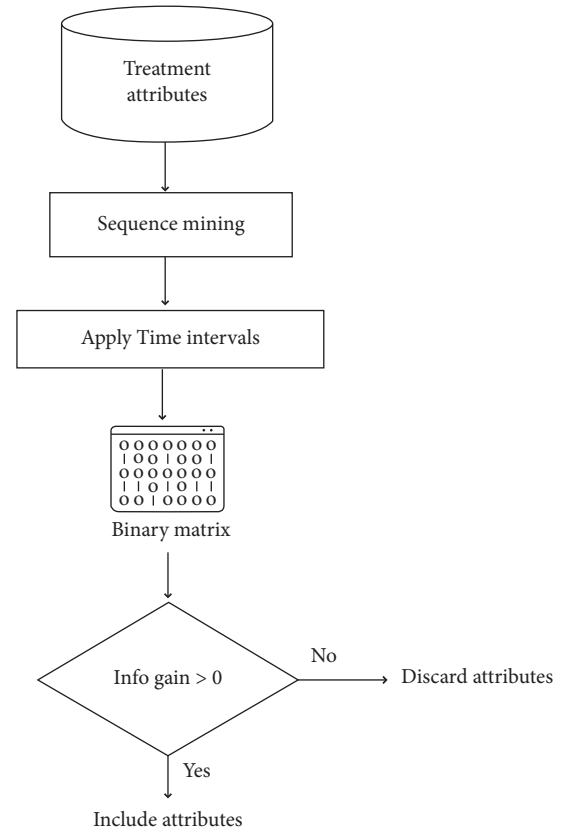


FIGURE 2: Treatment preprocessing.

$$\text{Info}(D) = - \sum_{i=1}^m p_i \log_2(p_i), \quad (1)$$

$$\text{Info}_A(D) = \sum_{j=1}^v \frac{D_j}{D} \times I(D_j), \quad (2)$$

$$\text{Gain}(A) = \text{Info}(D) - \text{Info}_A(D). \quad (3)$$

*Comorbidity Data Preprocessing.* The comorbidities were collected for each patient as to whether she has a particular condition or not. Comorbidities like chronic obstructive pulmonary disease (COPD), diabetes, hypertension, and coronary artery disease (CAD) were recorded and correspondingly, a metric-CCI was computed for each patient. Charlson Comorbidity Index (CCI) [34] calculates a person's ten-year mortality probability by administering assigned weights to different comorbidities. The higher the computed index, the higher the probability of mortality. For instance, a person having COPD gains +1 score in his/her CCI score. Similarly, patients with uncomplicated diabetes gain an additional +1, while an end organ damaged diabetes gains +3 score their CCI score. Thus, CCI was calculated for each patient to understand the effect of comorbidities better. A summary measure such as CCI is as good as comorbidities used to compute it. Its significance in prognosis has also been proven in the past [35]. Together with the performance



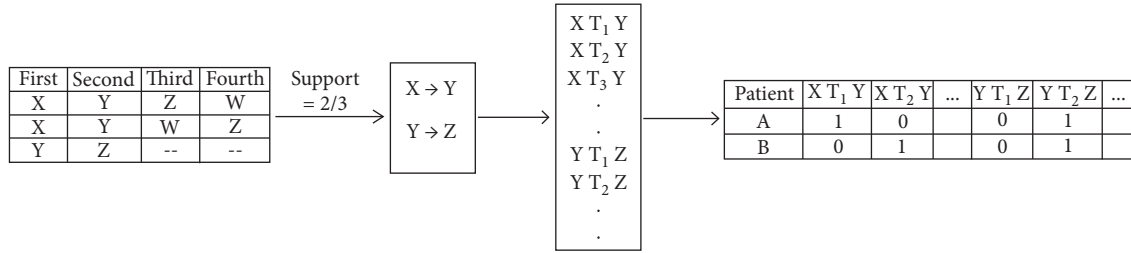


FIGURE 3: Time intervals in treatment sequences.

status values, CCI constitutes the quality of life part of the dataset in our study. Healthcare based IoT (IoHT) can be further utilized in collecting such life quality data in future studies [36].

**4.2.2. Data Summarization and Analysis.** The final set of different attributes and their description is shown in Table 1. The final dataset used for the analysis consists of 140 patients with a survival rate of 42.14% (59) and the dataset with a low degree of imbalance does not affect the predictors' performance [37]. Thus, no data balancing techniques have been employed in the study.

The present study analyzes survival based on some of the significant attributes and is shown in Figure 4.

Age has always been a controversial factor in the diagnosis and survival outcome for patients. In the present study, it is also revealed that, in younger age groups, patients have better survival outcomes than the older age group patients. However, unlike previous studies [38], ascites' presence has a somewhat opposite effect on advanced ovarian cancer patients' survival outcomes. In our dataset, patients having ascites have slightly better survival than the ones with no ascites present. Nonetheless, the existing literature did not consider the effect of ascites specifically in advanced stage. This result can be further examined by recording and assessing the ascites' volume present in future studies. CCI and ECOG, on the other hand, give promising analyses of survival outcomes. It can be seen from Figure 4 that the higher the values of CCI and ECOG, the lower the survival rate of the patients. ECOG graph shows a sharp declining trend in the chart except at ECOG performance status value 4. This slight change in the graph is that the number of patients with ECOG status 4 was only five, and the survival rate was 0%.

Similarly, a slight increase and inconsistency in the survival rate for patients with CCI score 6 are due to the small percentage of patients in that group. Thus, it is revealed from this consideration that patients with higher age, more comorbidities, and lower level of patient's general functioning are associated with lower survival rate. Other attributes like CA 125, histology, and grade did not show any relevant assessment and were not included in this study.

**4.3. Classification.** The integrated, processed data is supplied to classify the data into survived/deceased class. Ensemble techniques have been successfully used in various medical

datasets, and thus their applicability has been tested in the present study. A statistical method, logistic regression, has also been used for comparison with the ensemble approaches.

Bagging and boosting are ensemble classifiers. Bagging or Bootstrap aggregating creates  $k$  bootstrap sample datasets from the input dataset. Each test instance is classified using various base classifiers, and a combined classifier is created based on each base classifier's votes. The test instance is predicted with the class having majority votes. The averaging factor of voting helps in reducing any kind of variance in the dataset [19]. If the variance of a prediction is  $\sigma^2$ , then the variance of the average of  $k$  independent predictions is reduced to  $\sigma^2/k$ . However, boosting has a weighted average effect. Boosting boosts the performance by giving more importance to instances that are difficult to classify. If a classifier incorrectly classifies an instance, the next classifier provides more significance. Thus, boosting increases that instance's weight. Boosting performs better with weak classifiers as it reduces the bias that could not be removed with bagging. Nevertheless, we may face overfitting in boosting having a weighted approach. In the present study, AdaBoost is a type of boosting algorithm and has been utilized to classify the dataset. Assuming  $\text{err}(X_j)$  to be the misclassification error of tuple  $X_j$ , then the classifier  $M_i$  error rate is the sum of the weights of the misclassified tuples as given in equation (4). The weight of a classifier  $M_i$ 's vote will be as given in equation (5):

$$\text{error}(M_i) = \sum_j^d w_j \times \text{err}(X_j), \quad (4)$$

$$\log \frac{1 - \text{error}(M_i)}{\text{error}(M_i)}. \quad (5)$$

It has been confirmed from the previous studies that ensemble techniques, especially bagging and boosting, can perform better than most of the base classifiers individually.

When the base classifiers used for bagging are all decision tree classifiers, it is known as random forests. The forest denotes the collection of trees into a single unit (combined classifier). Random forest is called random as the decision trees are created using a random selection of attributes to decide the split at each node [39]. Each decision tree votes to determine the class of an instance, and the class with the majority votes is assigned to the test instance [40]. The random forest has proven to give better results in medical datasets. Another popular approach, called

TABLE 1: Dataset description.

	Attribute	Description	Range/values
Clinical attributes	Age	Age at the time of diagnosis	17-80 (median: 54)
	CA-125	CA-125 value at the time of diagnosis	8.7-16301 (median: 929.13)
	Ascites	Presence of ascites in the body	Yes: 114 No: 26
	Grade	Abnormality level of cancer cells	2-4 (median: 3)
	Stage	Figo substage	3-4 (median: 4) Clear cell: 1 Endometrioid: 4 Serous: 111 Small cell:1 Germ cell: 1 Mucinous: 6 Poorly/undifferentiated: 13 Mixed: 3
	Histology	Microscopic regularity of cancer cells	
Treatment attributes	Treatment sequences	Frequent treatment sequences obtained after sequence mining	Surgery → chemotherapy NACT → surgery NACT → hormonal therapy Chemotherapy → hormonal therapy Surgery → hormonal therapy Chemotherapy → CRS Surgery → NACT
Life quality attributes	CCI	Charlson comorbidity index obtained using comorbidities	2-9 (median: 3)
	ECOG performance status	The general well-being of a patient	1-5 (median: 2)
Class attribute	Outcome	Survival outcome after three years of cancer diagnosis	Yes: 59 No: 81

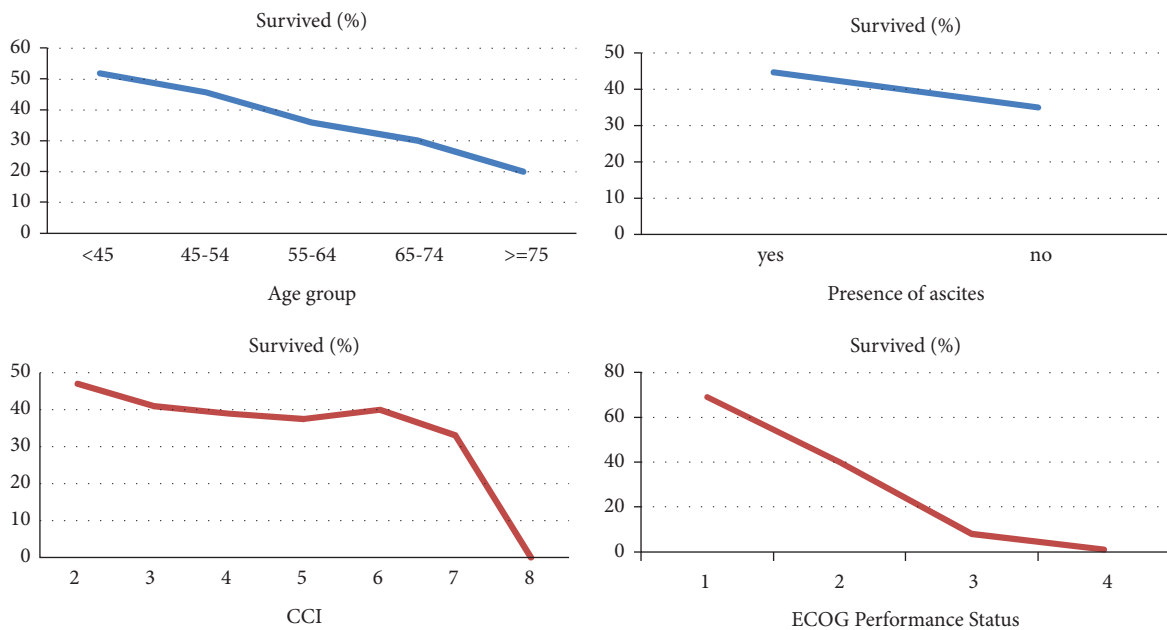


FIGURE 4: Data analysis with survival.

XGBoost, has been applied to the dataset using scikit-learn framework. It is a gradient tree boosting approach designed mainly to boost the speed and performance [41]. XGBoost can be used for both classification and regression problems.

It is a widely used algorithm by the researchers, specifically for scalable problems [42].

Logistic regression, being a statistical technique, has been used in the present study to compare ensemble

techniques with statistical data mining techniques. It models the class membership probability concerning the different attributes of the dataset. It divides the dataset into two classes based on the likelihood of each instance belonging to a particular class. The probability is computed with the help of the attributes' values and estimated coefficients for each feature. The attributes are assumed to be independent to give better performance results. Logistic regression has been used by various authors in different healthcare applications to aid the diagnosis or prediction processes [43, 44].

When some base classifiers like decision trees, SVM, and k-NN were used for the classification process, decision trees gave the best performance for the dataset. The performance of decision trees is acknowledged in various applications due to their high results [45, 46]. Decision trees are also preferred and explored in many medical applications because of their simple and better clarity to the clinicians [47]. Explainable AI is yet another domain that can be explored in future studies. Thus, the classifiers used in this study utilized decision trees for creating the ensemble. Since the dataset used had a small number of instances, 10-fold cross-validation was used for each technique. It divides the dataset into ten equal-sized partitions, where onefold acts as the testing partition, and all the other nine partitions are used for training the classifier. Thus, onefold is treated as a testing partition, and the method is fit on the remaining 9-1-fold. The results on testing partitions of 10-fold cross-validation are averaged. Classification was performed using Classification Learner app on MATLAB software. The final experimental details with highest performance of each classifier used are as illustrated in Table 2.

## 5. Result Analysis and Discussion

Table 3 shows the results for the time interval sequence mining approach. Time ranges of two months and six months achieved the highest accuracy; thus, Table 3 shows the evaluation measures for only 2 and 6 months. Boosting achieved the best results for both the cases in terms of accuracy and AUC. ROC curves are shown in Figure 5. 5-fold and 15-fold cross-validation techniques were also applied to the dataset to evaluate the methodologies. However, 10-fold gave better results than the other two validation techniques, with the highest of 72.9% for 5-fold and 75.4% for 15-fold. Also, as noted in the previous studies [48], ensemble techniques performed relatively better than the statistical method for our current research as well. However, it can be seen that time intervals of 2 months can better predict the survival of ovarian cancer in almost all the evaluation measures. Six-month approach gave slightly better results in specificity when only boosting is considered.

Contrary to the previous study using six months of time intervals for prostate cancer [33], we have evaluated different time intervals in our study to assess the appropriate time interval for advanced ovarian cancer patients. Thus, time intervals may vary according to cancer type as medications and cancer management differ for each cancer type. An appropriate model may have to be created according to the cancer type and possibly nature of cancer.

TABLE 2: Experimental details.

Model	Parameter settings
Bagging	Method = decision trees
	Max number of splits = 139
	Learning rate = 0.1
Boosting	Ensemble method = AdaBoost
	Max number of splits = 20
	Learning rate = 0.1
Random forest	Random number seed = 0
	Maximum depth = unlimited
XGBoost	Maximum number of trees = 100
Logistic regression	—

The treatment attributes selected for 2 months and 6 months are as given in Table 4. It is further revealed from the Table that the hypothesis in the present study that 6 months' interval might not be useful in the ovarian cancer dataset is true. The attributes selected in 6 months' intervals are only two, with both having T1 (0-6 months) intervals. Conversely, the attributes selected in 2 months' intervals are four and having varying intervals from T1 to T5 only. Thus, it might be possible that only a few, if any, patients received the next line of treatments after say 8-10 months of the previous treatments and, consecutively, did not have any significant role in survival prediction.

Further, each data profile's significance is examined by applying classification techniques to each category of attributes separately for a 2-month time interval integrated dataset. Table 5 shows the accuracy (in %) along with the classification technique for each data profile.

It is revealed from the results that when an individual category is considered, the life quality dataset performs better than the other data profiles. Also, the treatment dataset gave better accuracy than the clinical dataset. Thus, the treatments' sequences and the time elapsed in the treatments can give the clinicians and patients better knowledge of patients' survival outcomes. This result contributes to the current understanding that, for advanced ovarian carcinoma patients, clinical attributes like CA-125, grade, etc. can indicate selecting the appropriate treatment for the patient. Still, it might not be a good indicator for survival prediction of the patient. Nonetheless, treatment sequences and mostly life quality attributes can be better used in predicting survival outcome.

### 5.1. Comparison of Proposed Work with Existing Literature.

To determine the importance of sequence and time between different therapies given to a patient, we have further compared the proposed approach without sequence mining. A binary matrix has been created for the same, based on the medications received by each patient, irrespective of the sequence in which she received the therapy. An example of such a matrix is shown in Figure 6.

The comparison of various evaluation measures for all the approaches is shown in Table 6. Here, time interval approach attained better results than without sequence approach in all the criteria. However, specificity is the same

TABLE 3: Classification results.

		Accuracy (%)	True positive rate or sensitivity	Specificity	Area under curve
6 months	Bagging	71.4	<b>0.79</b>	0.61	0.80
	Random forest	70.7	0.64	0.8	0.72
	Boosting	<b>73.6</b>	0.69	<b>0.8</b>	<b>0.81</b>
	Logistic regression	65.7	0.68	0.63	0.70
	XGBoost	71.42	0.71	0.64	0.78
2 months	Bagging	74.3	<b>0.85</b>	0.59	0.82
	Random forest	75.7	0.72	<b>0.81</b>	0.82
	Boosting	<b>76.4</b>	0.80	0.71	<b>0.85</b>
	Logistic regression	67.1	0.64	0.71	0.70
	XGBoost	73.8	0.73	0.63	0.79

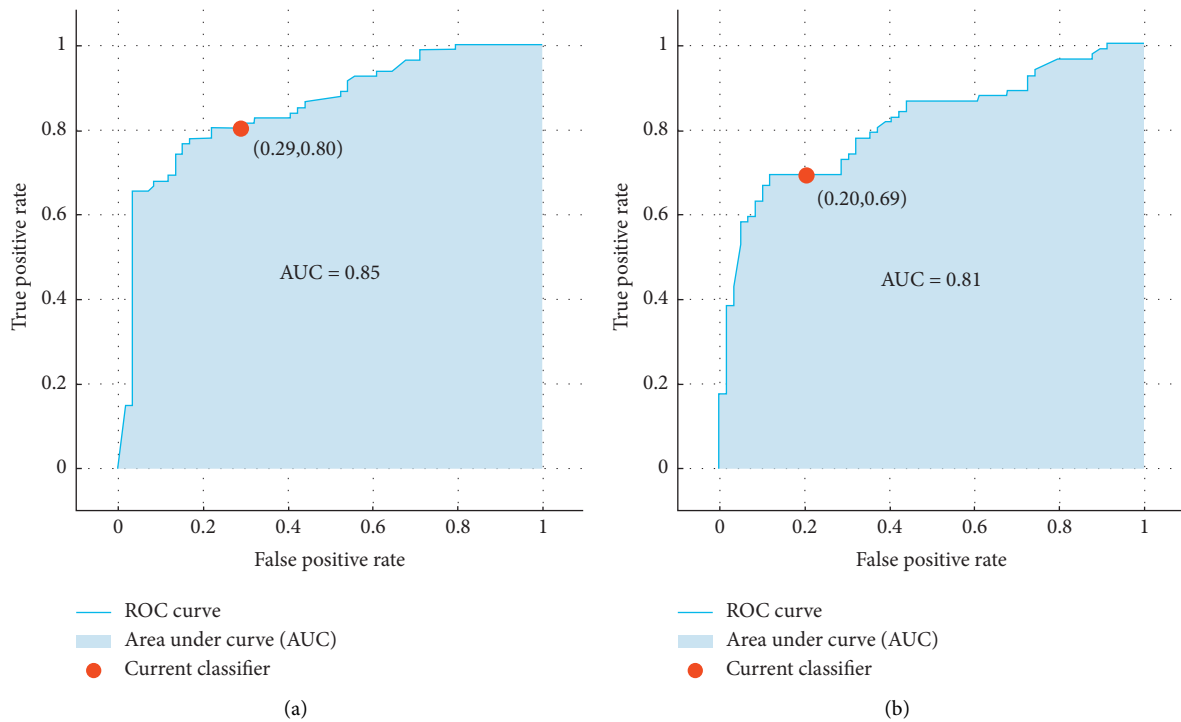


FIGURE 5: ROC curves for (a) boosting in 2 months’ time interval; (b) boosting in 6 months’ time interval.

TABLE 4: Treatment attributes selected.

2 months’ time interval		6 months’ time interval	
Attributes	Information gain	Attributes	Information gain
Chemotherapy_T5_CRS	0.0458	Chemotherapy_T1_hormonal therapy	0.0408
Surgery_T5_chemotherapy	0.0272	NACT_T1_hormonal therapy	0.008
Chemotherapy_T4_CRS	0.023		
NACT_T1_hormonal therapy	0.01		

TABLE 5: Classification results for each data profile.

Data profile	Highest accuracy in % (classifier)
Clinical dataset	61.4 (bagging)
Treatment dataset	65 (boosting)
Life quality dataset	71.4 (boosting)

Patient	First	Second	Third	Fourth	→	Patient	Treatment X	Treatment Y	Treatment Z	Treatment W
A	X	Y	Z	--		A	1	1	1	0
B	X	Y	W	--		B	1	1	0	1

FIGURE 6: Without sequence treatment processing.

TABLE 6: Comparison of results.

	Accuracy	Sensitivity or true positive rate	Specificity	Area under curve
Without sequence mining	0.707	0.78	0.71	0.77
2-month time interval	<b>0.764</b>	<b>0.80</b>	0.71	<b>0.85</b>
6-month time interval	0.736	0.69	<b>0.8</b>	0.81

for two-month time interval and no-sequence mining approach. But the overall results improved in the time interval approach. The graphical representation of the results is shown in Figure 7. Also, the random forest gave better results than bagging and boosting in without sequence mining approach. Here, also, the parameter settings for random forest were the same as in the case of proposed approach (i.e., random number seed=0 and maximum depth = unlimited). Thus, the results for only random forest have been presented in the results. The significance of time in specific treatments has also been acknowledged in previous literature on advanced epithelial ovarian cancer [13]. Hofstetter et al. [49] also demonstrated the use of intervals between surgery and chemotherapy in advanced ovarian cancer patients using statistical techniques. They also revealed that the periods were around 3–6 weeks. The present study also gave better results when time intervals of 2 months were used for the survival prediction. The results were validated statistically by computing  $t$ -score and corresponding  $p$ -values with a significance level of 0.05. Since the best results were given by 2-month time interval approach, it was compared with ‘without sequence mining’ approach. The results are given in Table 7, and it is evident that the result is significant at  $p < 0.05$ .

We have additionally generated an assessment of some of the recent studies to compare the data profiles and techniques used in the present study with existing literature, given in Table 8. It can be observed from Table 8 that the majority of the studies used only clinical and treatment data for survival prediction, where treatment data mostly includes details of primary treatment only. Malhotra et al. [56] used treatment sequences along with clinical and genetic data, though the authors did not consider the time elapsed between the treatments. Also, it can be seen from Table 5 that life quality data has a significant contribution to the survival prediction, which is lacking in [56]. The collection and analyses of genetic data, however, can be the future work for the study. Studies using neural networks and deep learning are also becoming more common now with genetic and multimodal data and thus, can be utilized in future studies [62–64]. These have been further explored in various image based datasets as well for the detection and prediction purposes [65–67]. Deep learning technology has been proven in various studies to outperform basic machine learning techniques [68, 69]. However, the dataset in the

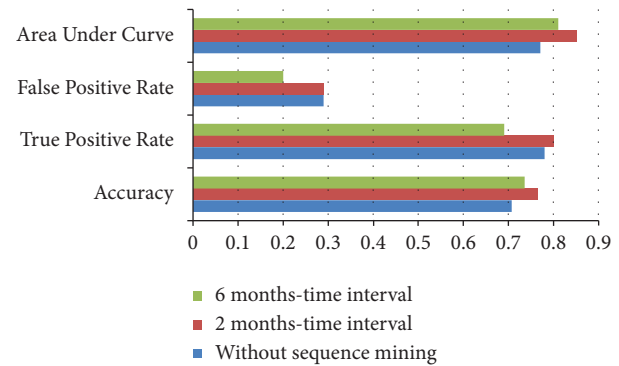


FIGURE 7: Comparison of results.

TABLE 7: Statistical significance.

Approach	‘2-months time interval’ with ‘without sequence mining’
$t$ -value	1.90429
$p$ value	0.036491

present study has a smaller number of instances than the existing literature, and deep learning can perform better with large amounts of training data. Due to lack of significant training data, deep learning could not be explored in this study. But this is because the present study involves recent records and only advanced-stage patients. Since earlier stages of almost all cancer already have around 90% survival rates, survival prediction is an easier task. But in the later stages, the survival rates vary from about 10% to 40%. Thus, the present study creates a model established on the cancer behavior (for advanced stage only) that will be more useful for clinicians in examining the survival of cancer patients [70]. It can be observed from Table 8 that almost all the studies used dataset of all the stages. Guo et al. [57] considered earlier cancer patients for the survival prediction and achieved high results. However, as already discussed, earlier stages have considerably higher survival rates and is mostly easier to predict. Thus, more research on advanced cancer patients needs to be conducted to further compare the results.

Thus, it can be concluded from the results that the time interval approach gave better results than no-sequence approach. The time intervals may vary, but the time between

TABLE 8: Comparison of techniques with previous literature.

S.no.	Authors	Dataset	Type of cancer with stage	Stage of cancer patients used	Type of attributes	Classification technique used	Results
1.	Matsuo et al. [50]	Clinical-768 patients	Cervical cancer	All stage	(i) Clinical (ii) Treatment	Deep learning and cox proportional model	Mean absolute error of 30.7 (deep learning), 43.6 (cox proportional hazard regression)
2.	Park et al. [51]	SEER dataset	Breast cancer	All stage	(i) Clinical (ii) Treatment	Subgroup mining	Effective rules generated
3.	Simsek et al. [29]	SEER dataset	Breast cancer	All stage	(i) Clinical	ANNs and logistic regression	83.6% (ANNs) 82.9% (LR) for 5-year survival
4.	Wang et al. [52]	Clinical-1075 patients	Lung cancer	All stage	(i) Clinical (ii) Treatment (iii) Comorbidities	Gaussian bayesian network	$R^2$ of 93.57% (stage-I), 86.83% (stage-II), 67.22% (stage-III), 52.94% (stage-IV)
5.	García-Laencina et al. [53]	Clinical-399 patients	Breast cancer	All stage	(i) Clinical (ii) Treatment	KNN, logistic regression, decision trees, support vector machine	81% (highest in KNN)
6.	Toth et al. [54]	National health database-28817 patients	Colon cancer	All stage	(i) Treatment	Sequence mining	—
7.	Koo et al. [30]	Clinical-7267 patients	Prostate cancer	All stage	(i) Clinical (ii) Treatment	Artificial neural networks	84.9% overall 5-year survival
8.	Kate and Nadig [55]	SEER dataset	Breast cancer	All stage	(i) Clinical (ii) Treatment	Logistic regression, naïve bayes, decision tree	84.2% (naïve bayes)
9.	Malhotra et al. [56]	Clinical-393 patients	Glioblastoma cancer	All stage	(i) Treatment (ii) Genetic (iii) Clinical	Sequence mining with statistical techniques	85% (logistic regression)
10.	Guo et al. [57]	Clinical-5842 patients	Cervical cancer	Stage IA1 to IIB2	(i) Clinical	SVM, decision tree, random forest, ANN etc.	0.895 and 0.89 AUC (light GBM and random forest)
11.	Kalafi et al. [58]	University Malaya medical cancer registry-8066 patients	Breast cancer	All stage	(i) Clinical (ii) Treatment	SVM, MLP (multilayer perceptron), decision trees, random forest	88.2% accuracy (MLP)
12.	Alabi et al. [59]	SEER dataset	Oral cancer	All stage	(i) Clinical	Logistic regression, SVM, bayes point, boosting, decision forest, decision jungle	88.7% (boosting)
13.	Bos et al. [60]	Clinical-177 patients	Oral cancer	All stage	(i) Clinical (ii) Radiomic (MRI)	Logistic regression	0.744 AUC
14.	Hira et al. [61]	TCGA-579 and 593 samples	Ovarian cancer	All stage	(i) Multi-omics data	Deep learning	93.2–95.5% and 87.1–95.7% accuracy
15.	Proposed approach	Clinical-140 patients	Ovarian cancer	Advanced stage	(i) Clinical (ii) Treatment (iii) Life quality (comorbidities + ECOG)	Sequence mining with ensemble	76.4% accuracy and 0.85 AUC (boosting)

treatments can also create a better and reliable predictive model for other cancer patients. The integrated dataset, including data from all profiles, is a better prediction model than the existing models, including only clinical attributes and treatment attributes with no frequent sequences. The clinicians can use this information while deciding the appropriate treatments for advanced ovarian carcinoma patients and the elapsed time between each treatment. The patients' general well-being can also be useful indicators in determining the treatments and corresponding overall survivability of the patients.

## 6. Conclusion

Advanced ovarian carcinoma patients have a poor prognosis compared to early-stage patients. The present study gives some worthwhile comprehensions in advanced ovarian cancer survival. An integrated predictive model has been created using three different data profiles from a real-world clinical dataset. It also focuses on the significance of treatment sequences with varying time elapsed between treatments and various life quality attributes in the survival analysis of patients. Cancer patients are often treated with multiple lines of therapy. The present study validates and ascertains the use of varying time elapsed between treatments in examining the survival of patients using a modified sequential mining algorithm of GSP, and various machine learning techniques. It was revealed that life quality attributes and treatment sequences with the time intervals could predict survival better than clinical facts. Also, time intervals of two months between the treatment sequences performed better than other time intervals with an accuracy of 76.4% and 0.85 AUC. The proposed approach of modified sequential mining algorithm and classification with 76.4% accuracy performed better than the existing approach without sequential mining, giving around 70% accuracy. The results were also statistically validated. Thus, the clinicians and researchers should consider patients' quality of life and line of treatments with time elapsed between them while creating a predictive model for cancer patients.

However, there are a few limitations and possible future aspects worth noting. This study used data from only five years of the hospital to record current medications and other medical technologies. The dataset thus had a small number of instances, which could have resulted in overfitting in classification. Also, the dataset was collected manually by the authors. Some recording errors might have been created in the data. Besides, precise medications and chemotherapy cycles or dosage were not considered to avoid confusion. Yet, these might be useful if we had a larger dataset.

## Data Availability

The study was made possible with the dataset collected from Rajiv Gandhi Cancer Institute & Research Center, New Delhi.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

The authors acknowledge and extend their appreciation to the Researchers Supporting Project number (RSP-2021/124), King Saud University, Riyadh, Saudi Arabia, for funding this study.

## References

- [1] *The Top 10 Causes of Death*, WHO, Geneva, Switzerland, Available at: <https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death>, 2020.
- [2] R. J. Kate and R. Nadig, "Stage-specific predictive models for breast cancer survivability," *International Journal of Medical Informatics*, vol. 97, pp. 304–311, 2017.
- [3] S. Walczak and V. Velanovich, "Improving prognosis and reducing decision regret for pancreatic cancer treatment using artificial neural networks," *Decision Support Systems*, vol. 106, pp. 110–118, 2018.
- [4] K. Juneja and C. Rana, "An improved weighted decision tree approach for breast cancer prediction," *International Journal of Information Technology*, vol. 12, no. 3, pp. 797–804, 2020.
- [5] H. M. Zolbanin, D. Delen, and A. Hassan Zadeh, "Predicting overall survivability in comorbidity of cancers: a data mining approach," *Decision Support Systems*, vol. 74, pp. 150–161, 2015.
- [6] "Surveillance, epidemiology, and end results (SEER) Program," 1975, <https://www.seer.cancer.gov>.
- [7] T. R. Network, *The Cancer Genome Atlas Data Portal*, National Institute of Health, Maryland, USA, 2010.
- [8] M. Z. Nezhad, N. Sadati, K. Yang, and D. Zhu, "A Deep Active Survival Analysis approach for precision treatment recommendations: application of prostate cancer," *Expert Systems with Applications*, vol. 115, pp. 16–26, 2019.
- [9] C. A. Hamilton, A. Miller, Y. Casablanca et al., "Clinicopathologic characteristics associated with long-term survival in advanced epithelial ovarian cancer: an NRG Oncology/Gynecologic Oncology group ancillary data study," *Gynecologic Oncology*, vol. 148, no. 2, pp. 275–280, 2018.
- [10] L. Vincent, C. Jankowski, L. Ouldamer et al., "Prognostic factors of overall survival for patients with FIGO stage IIIC or IVa ovarian cancer treated with neo-adjuvant chemotherapy followed by interval debulking surgery: a multicenter cohort analysis from the FRANCOGYN study group," *European Journal of Surgical Oncology*, vol. 46, no. 9, pp. 1689–1696, 2020.
- [11] J. S. Dungan, "Neoadjuvant chemotherapy or primary surgery in stage IIIC or IV ovarian cancer," *Year Book of Obstetrics, Gynecology, and Women's Health*, vol. 2011, pp. 507–508, 2011.
- [12] S. Akhavan, A. Ghahghaei-Nezamabadi, M. Modaresgilani et al., "Impact of diabetes mellitus on epithelial ovarian cancer survival," *BMC Cancer*, vol. 18, no. 1, 2018.
- [13] A. Maheshwari, N. Kumar, S. Gupta et al., "Outcomes of advanced epithelial ovarian cancer treated with neoadjuvant chemotherapy," *Indian Journal of Cancer*, vol. 55, no. 1, pp. 50–54, 2018.
- [14] Y. Zhang, G. Luo, M. Li et al., "Global patterns and trends in ovarian cancer incidence: age, period and birth cohort analysis," *BMC Cancer*, vol. 19, no. 1, p. 984, 2019.

- [15] S. B. Coburn, F. Bray, M. E. Sherman, and B. Trabert, "International patterns and trends in ovarian cancer incidence, overall and by histologic subtype," *International Journal of Cancer*, vol. 140, no. 11, pp. 2451–2460, 2017.
- [16] R. Takiar, "Status of ovarian cancer in India (2012–14)," *EC Gynaecology*, vol. 8, no. 5, pp. 358–364, 2019.
- [17] N. Bhatla, "The world ovarian cancer coalition atlas: global trends in incidence, mortality and survival," 2018.
- [18] S. Sato and H. Itamochi, "Neoadjuvant chemotherapy in advanced ovarian cancer: latest results and place in therapy," *Therapeutic advances in medical oncology*, vol. 6, no. 6, pp. 293–304, 2014.
- [19] M. J. Zaki and W. Meira, *Data Mining and Analysis*, Cambridge University Press, Cambridge, UK, 2014.
- [20] R. Srikant and R. Agrawal, "Mining sequential patterns: generalizations and performance improvements," in *Proceedings of the 5th International Conference on Extending Database Technology: Advances in Database Technology (EDBT '96)*, P. M. G. Apers, M. Bouzeghoub, and G. Gardarin, Eds., pp. 3–17, Springer-Verlag, London, UK, 1996.
- [21] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proceedings of the 20th International Conference on Very Large Data Bases, VLDB*, Santiago, Chile, September 1994.
- [22] M. J. Zaki, "SPADE: an efficient algorithm for mining frequent sequences," *Machine Learning*, vol. 42, no. 1/2, pp. 31–60, 2001.
- [23] P. Jian Pei, H. Jiawei Han, B. Mortazavi-Asl et al., "PrefixSpan: mining sequential patterns efficiently by prefix-projected pattern growth," in *Proceedings of the 17th International Conference on Data Engineering*, Heidelberg, Germany, April 2001.
- [24] K. Deng, C. Yang, Q. Tan et al., "Sites of distant metastases and overall survival in ovarian cancer: a study of 1481 patients," *Gynecologic Oncology*, vol. 150, no. 3, pp. 460–465, 2018.
- [25] J. S. Hildebrand, K. Wallace, W. S. Graybill, and L. E. Kelemen, "Racial disparities in treatment and survival from ovarian cancer," *Cancer Epidemiology*, vol. 58, pp. 77–82, 2019.
- [26] M. P. Viswanathan, D. S. Kumar, P. Muniasamy, S. Navin Noushad, and D. Pradeep, "Patterns of relapse and survival analysis of advanced epithelial ovarian cancers operated in a tertiary cancer centre," *Indian Journal of Gynecologic Oncology*, vol. 17, p. 72, 2019.
- [27] C.-J. Tseng, C.-J. Lu, C.-C. Chang, G.-D. Chen, and C. Cheewakriangkrai, "Integration of data mining classification techniques and ensemble learning to identify risk factors and diagnose ovarian cancer recurrence," *Artificial Intelligence in Medicine*, vol. 78, pp. 47–54, 2017.
- [28] M. Lu, Z. Fan, B. Xu et al., "Using machine learning to predict ovarian cancer," *International Journal of Medical Informatics*, vol. 141, Article ID 104195, 2020.
- [29] S. Simsek, U. Kursuncu, E. Kibis, M. AnisAbdellatif, and A. Dag, "A hybrid data mining approach for identifying the temporal effects of variables associated with breast cancer survival," *Expert Systems with Applications*, vol. 139, Article ID 112863, 2019.
- [30] K. C. Koo, K. S. Lee, S. Kim et al., "Long short-term memory artificial neural network model for prediction of prostate cancer survival outcomes according to initial treatment strategy: development of an online decision-making support system," *World Journal of Urology*, vol. 38, no. 10, pp. 2469–2476, 2020.
- [31] M. N. Doja, I. Kaur, and T. Ahmad, "Age-specific survival in prostate cancer using machine learning," *Data Technologies and Applications*, vol. 54, no. 2, pp. 215–234, 2020b.
- [32] P. Fournier-Viger, J. C. W. Lin, A. Gomariz et al., "The SPMF open-source data mining library version 2," in *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, September 2016.
- [33] I. Kaur, M. N. Doja, and T. Ahmad, "Time-range based sequential mining for survival prediction in prostate cancer," *Journal of Biomedical Informatics*, vol. 110, Article ID 103550, 2020.
- [34] S. Schneeweiss, P. S. Wang, J. Avorn, and R. J. Glynn, "Improved comorbidity adjustment for predicting mortality in Medicare populations," *Health Services Research*, vol. 38, no. 4, pp. 1103–1120, 2003.
- [35] S. R. Austin, Y. N. Wong, R. G. Uzzo, J. R. Beck, and B. L. Egleston, "Why summary comorbidity measures such as the Charlson comorbidity index and elixhauser score work," *Medical Care*, vol. 53, no. 9, pp. e65–e72, 2015.
- [36] E. M. Abou-Nassar, A. M. Iliyasa, P. M. El-Kafrawy, O.-Y. Song, A. K. Bashir, and A. A. A. El-Latif, "DITrust chain: towards blockchain-based trust models for sustainable healthcare IoT systems," *IEEE Access*, vol. 8, 2020.
- [37] M. Hammad, M. H. Alkinani, B. B. Gupta, and A. A. Abd El-Latif, "Myocardial infarction detection based on deep neural network on imbalanced data," *Multimedia Systems*, 2021.
- [38] E. Kipps, D. S. Tan, and S. B. Kaye, "Meeting the challenge of ascites in ovarian cancer: new avenues for therapy and research," *Nature Reviews Cancer*, vol. 13, no. 4, pp. 273–8210, 2013.
- [39] C. C. Aggarwal, *Data Mining: The Textbook*, Springer International Publishing, New-York, NY, USA, 2015.
- [40] J. Han and M. Kamber, *Data Mining Concepts and Techniques*, 3rd edition, 2012.
- [41] T. Chen and C. Guestrin, "XGBoost," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining-KDD '16*, 2016.
- [42] M. Akcay, D. Etiz, and O. Celik, "Prediction of survival and recurrence patterns by machine learning in gastric cancer cases undergoing radiation therapy and chemotherapy," *Advances in radiation oncology*, vol. 5, no. 6, pp. 1179–1187, 2020.
- [43] T. Mazzocco and A. Hussain, "Novel logistic regression models to aid the diagnosis of dementia," *Expert Systems with Applications*, vol. 39, no. 3, pp. 3356–3361, 2012.
- [44] C. Ieracitano, N. Mammone, A. Bramanti, S. Marino, A. Hussain, and F. C. Morabito, "A Time-frequency based machine learning system for brain states classification via EEG signal processing," in *Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN)*, Budapest, Hungary, July 2019.
- [45] S. Höppner, E. Stripling, B. Baesens, S. vanden Broucke, and T. Verdonck, "Profit driven decision trees for churn prediction," *European Journal of Operational Research*, vol. 284, pp. 920–933, 2020.
- [46] I. Kaur, G. S. Narula, and V. Jain, "Differential analysis of token metric and object oriented metrics for fault prediction," *International Journal of Information Technology*, vol. 9, pp. 93–100, 2017.
- [47] K. Mathan, P. M. Kumar, P. Panchatcharam, G. Manogaran, and R. Varadharajan, "A novel Gini index decision tree data mining method with neural network classifiers for prediction of heart disease," *Design Automation for Embedded Systems*, vol. 22, pp. 225–242, 2018.



- [48] W. Książek, M. Hammad, P. Pławiak, U. R. Acharya, and R. Tadeusiewicz, "Development of novel ensemble model using stacking learning and evolutionary computation techniques for automated hepatocellular carcinoma detection," *Biocybernetics and Biomedical Engineering*, vol. 40, 2020.
- [49] G. Hofstetter, N. Concin, I. Braicu et al., "The time interval from surgery to start of chemotherapy significantly impacts prognosis in patients with advanced serous ovarian carcinoma-analysis of patient data in the prospective OVCAD study," *Gynecologic Oncology*, vol. 131, no. 1, pp. 15–20, 2013.
- [50] K. Matsuo, S. Purushotham, B. Jiang et al., "Survival outcome prediction in cervical cancer: cox models vs deep-learning model," *American Journal of Obstetrics and Gynecology*, vol. 220, no. 381, pp. e1–e14, 2019.
- [51] J. V. Park, S. J. Park, and J. S. Yoo, "Finding characteristics of exceptional breast cancer subpopulations using subgroup mining and statistical test," *Expert Systems with Applications*, vol. 118, pp. 553–562, 2019.
- [52] K. J. Wang, J. L. Chen, K. H. Chen, and K. M. Wang, "Survivability prognosis for lung cancer patients at different severity stages by a risk factor-based bayesian network modeling," *Journal of Medical Systems*, vol. 44, no. 3, p. 65, 2020.
- [53] P. J. García-Laencina, P. H. Abreu, M. H. Abreu, and N. Afonso, "Missing data imputation on the 5-year survival prediction of breast cancer patients with unknown discrete values," *Computers in Biology and Medicine*, vol. 59, pp. 125–133, 2015.
- [54] K. Tóth, I. Kósa, and Á. Vathy-Fogarassy, "Frequent treatment sequence mining from medical databases," *Studies in Health Technology and Informatics*, vol. 236, pp. 211–218, 2017.
- [55] R. J. Kate and R. Nadig, "Stage-specific predictive models for breast cancer survivability," *International Journal of Medical Informatics*, vol. 97, pp. 304–311, 2018.
- [56] K. Malhotra, S. B. Navathe, D. H. Chau, C. Hadjipanayis, and J. Sun, "Constraint based temporal event sequence mining for Glioblastoma survival prediction," *Journal of Biomedical Informatics*, vol. 61, pp. 267–275, 2016.
- [57] C. Guo, J. Wang, Y. Wang et al., "Novel artificial intelligence machine learning approaches to precisely predict survival and site-specific recurrence in cervical cancer: a multi-institutional study," *Translational Oncology*, vol. 14, no. 5, Article ID 101032, 2021.
- [58] E. Y. Kalafi, N. A. M. Nor, N. A. Taib, M. D. Ganggayah, C. Town, and S. K. Dhillon, "Machine learning and deep learning approaches in breast cancer survival prediction using clinical data," *Folia Biologica*, vol. 65, no. 5-6, pp. 212–220, 2019.
- [59] R. O. Alabi, A. A. Mäkitie, M. Pirinen, M. Elmusrati, I. Leivo, and A. Almangush, "Comparison of nomogram with machine learning techniques for prediction of overall survival in patients with tongue cancer," *International Journal of Medical Informatics*, vol. 145, Article ID 104313, 2021.
- [60] P. Bos, M. W. M. Van Den Brekel, Z. A. R. Gouw et al., "Improved outcome prediction of oropharyngeal cancer by combining clinical and MRI features in machine learning models," *European Journal of Radiology*, vol. 139, Article ID 109701, 2021.
- [61] M. T. Hira, M. A. Razzaque, C. Angione, G. Scrivens, S. Sawan, and M. Sarker, "Integrated multi-omics analysis of ovarian cancer using variational autoencoders," *Scientific Reports*, vol. 11, p. 6265, 2021.
- [62] C. Ieracitano, N. Mammone, A. Hussain, and F. C. Morabito, "A novel multi-modal machine learning based approach for automatic classification of EEG recordings in dementia," *Neural Networks*, vol. 123, pp. 176–190, 2020.
- [63] M. Mahmud, M. S. Kaiser, A. Hussain, and S. Vassanelli, "Applications of deep learning and reinforcement learning to biological data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2063–2079, 2018.
- [64] S. Poria, E. Cambria, A. Hussain, and G.-B. Huang, "Towards an intelligent framework for multimodal affective data analysis," *Neural Networks*, vol. 63, pp. 104–116, 2015.
- [65] A. Sedik, A. M. Ilyasu, B. Abd El-Rahiem et al., "Deploying machine and deep learning models for efficient data-augmented detection of COVID-19 infections," *Viruses*, vol. 12, no. 7, p. 769, 2020.
- [66] A. Sedik, M. Hammad, F. E. Abd El-Samie, B. B. Gupta, and A. A. Abd El-Latif, "Efficient deep learning approach for augmented detection of coronavirus disease," *Neural Computing and Applications*, 2021.
- [67] A. Alghamdi, M. Hammad, H. Ugail et al., "Detection of myocardial infarction based on novel deep transfer learning methods for urban healthcare in smart cities," *Multimedia Tools and Applications*, 2020.
- [68] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: a brief review," *Computational Intelligence and Neuroscience*, vol. 2018, Article ID 7068349, 13 pages, 2018.
- [69] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: a review," *Neurocomputing*, vol. 187, pp. 27–48, 2016.
- [70] M. N. Doja, I. Kaur, and T. Ahmad, "Current state of the art for survival prediction in cancer using data mining techniques," *Current Bioinformatics*, vol. 15, p. 174, 2020a.