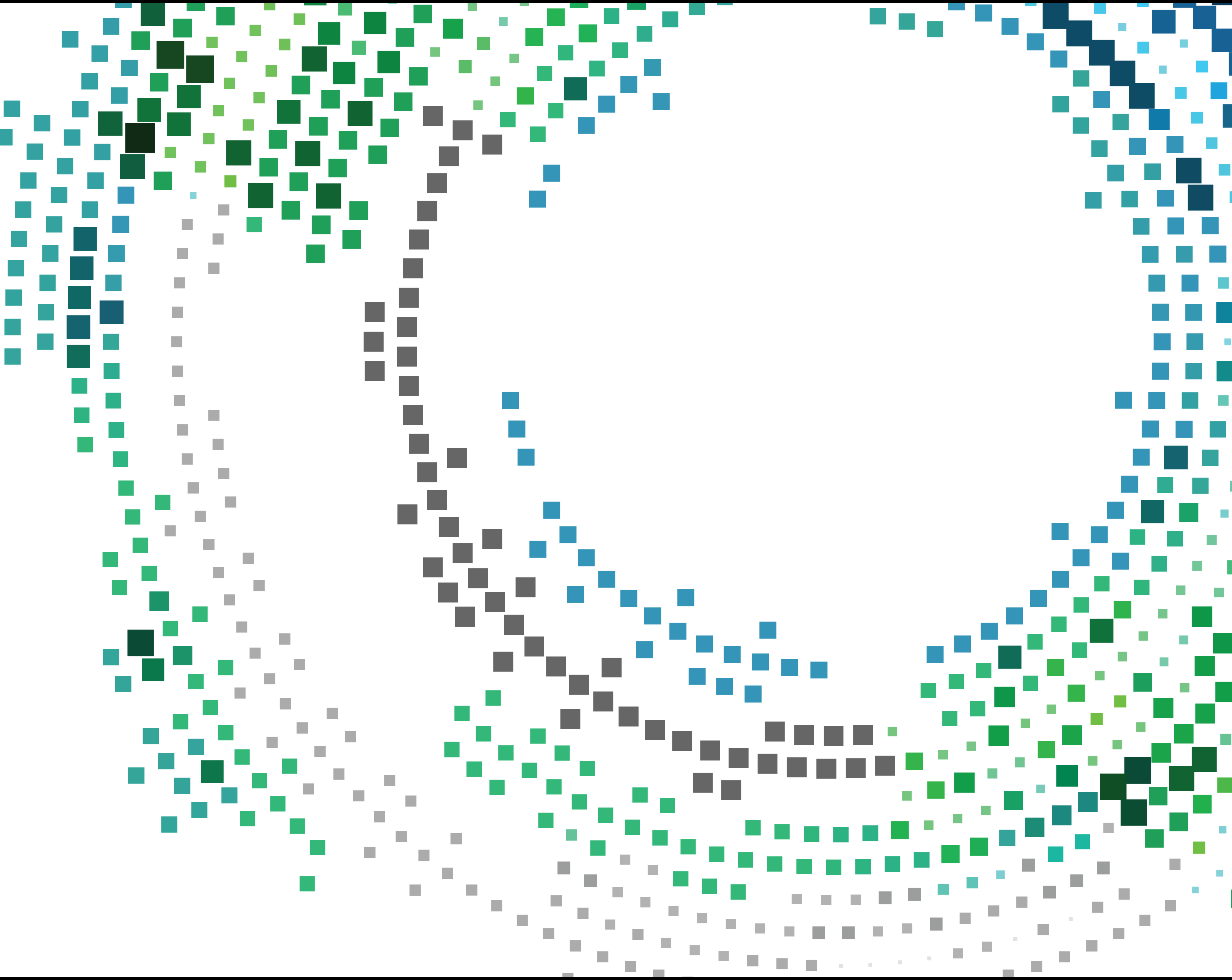
Mobile Information Systems

Intelligent Data Analytics for Internet of Things-Based Applications 2022

Lead Guest Editor: Omar Cheikhrouhou
Guest Editors: Habib Hamam, Abdelouahid Derhab, and Moez Krichen



Intelligent Data Analytics for Internet of Things-Based Applications 2022

Mobile Information Systems

Intelligent Data Analytics for Internet of Things-Based Applications 2022

Lead Guest Editor: Omar Cheikhrouhou
Guest Editors: Habib Hamam, Abdelouahid Derhab, and Moez Krichen

Chief Editor

Academic Editors

Contents

Hindawi
Mobile Information Systems
Volume 2022, Article ID 2123662, 12 pages
https://doi.org/10.1155/2022/2123662

Research Article

Hyper-Tuned CNN Using EVO Technique for Efficient Biomedical Image Classification

Mohammad Alnabhan,[1] **Ahmad Khader Habboush**,[2] **Qasem Abu Al-Haija**,[3] **Arup Kumar Mohanty**,[4] **Saumendra Pattnaik**,[4] **and Binod Kumar Pattanayak**[4]

[1] *Department of Computer Science, Princess Sumaya University for Technology (PSUT), Amman, Jordan*
[2] *Faculty of Information Technology, Jerash University, Jerash, Jordan*
[3] *Department of Cyber Security, Princess Sumaya University for Technology (PSUT), Amman, Jordan*
[4] *Department of Computer Science & Info. Technology, Institute of Technical Education & Research, Odisha, Bhubaneswar, India*

Correspondence should be addressed to Qasem Abu Al-Haija; q.abualhaija@psut.edu.jo

Received 14 April 2022; Revised 7 May 2022; Accepted 16 May 2022; Published 22 June 2022

Academic Editor: Abdelouahid Derhab

This research utilizes metaheuristic optimization inspired by the Egyptian Vulture Optimization (EVO) technique. Biomedical image segregation is developed to reduce the complex association of hyperparameters of Convolutional Neural networks (CNN). The complex attributes of CNN include the type of kernel, size of the kernel, size of the batch, epoch counts, momentum, learning rate, activation function, convolution layer, and dropout. However, the life cycle of an Egyptian vulture influences the optimization technique to resolve complexity and increase the accuracy of CNN. The proposed CNN-based EVO model was evaluated in comparison to ANN-based and deep learning-based classifiers utilizing brain MRI image datasets. The results achieved have confirmed the efficiency and performance of the proposed CNN-based EVO model, in which the average detection accuracy and precision were 93% and 95%, respectively.

1. Introduction

The heavy weightage of image data is readily available for the last few years regarding biomedical picture analysis and medical text practice. Nowadays, various new techniques have evolved based on machine learning techniques. The authentic analysis of images obtained from different clinical records by means of consistency, confidentiality, transparency, and validation of generated data is a challenging task. In the image analysis system, brain MRI image analysis is most trending in the research sector [1]. The biomedical data analysis has taken place before categorizing data for diagnostic analysis. Establishing such an efficient model in order to classify image selection is a crucial task for the machine learning approach. Recent works have taken place based on CNN-based classifications due to its abundant quality of extracting features [2]. The use of CNN requires a set of parameters that function manually after tuning to the data externally. These parameters help in building an efficient and accurate model in a particular time window. So, these are popularly known as hyperparameters. These parameters constitute the variables of the network structure of CNN. These parameters are best in use for image classification when categorized without external source intervention [3]. Many optimization techniques such as genetic algorithms and evolutionary algorithms have evolved as a fruitful solution to optimize hyperparameters. But most of the recent research works have focused on optimization techniques. The lacuna in recent works is tuning the hyperparameters with CNN-based models and optimizing the assumptions for validation [4]. This is a practical scenario in the medical sector where image classification is done best by authenticating the information and classifying it into objects. Therefore, in this work, we have focused on hyperparameter selection based on tuned CNN architecture, which is optimized for image classification [5]. We have considered and implemented Egyptian Vulture Optimization (EVO), which is a nature-inspired algorithm to get the optimized value for the best suit of the optimization technique.

The use of gradient-based optimization machine learning technique was complex for mathematical calculations consisting of limited tools of elements. But these methods are inconsistent in terms of exact function calculation and efficient for limited tools. To avoid such kinds of limitations, the metaheuristic optimization techniques come into a picture that is cost-friendly, and simple for calculation and experiment. The biological features of vultures are considered in this work mathematically to tune the hyperparameters of CNN [6].

The EVO-based algorithm explains the section process of hyperparameters tuned into the CNN network by optimizing the variables of parameters obtained by the training procedure. The variables of parameters are assigned by means of the upper and lower bound of the vector. Mathematically, these variables are modeled in order to get the best-optimized value of the hyperparameter. EVO technique is used in order to get the accurate optimized value in terms of hybridized exploitation and exploration of biological features extracted from this metaheuristic optimization technique.

A huge number of researches on biomedical image segregation for designing the CAD-based image classification system are done by researchers nowadays [1]. It deals with research on the classification of MRI images and many other biomedical images [7].

The exact execution and performance analysis of data in machine learning algorithms scenario is a widely used solution for all types of data analysis issues. But it is important to note that the design of data analysis is taken place using machine learning techniques by selecting the tools and models to obtain optimal values. Parameters are selected manually to process the learning phase and choose the classifier to get the optimal values. It is vital to note that the parameters are basically of two types, which are model parameters and hyperparameters. Model parameters are weighted neurons of the ANN model and get updated by the learning process [1]. The hyperparameters are designed to train the ANN model from the learning phase to design the model. Recent works have shown that a model of tuned hyperparameters can achieve a finite result of improved performance. A multispectral method based upon CNN has been designed to get the optimal values of hyperparameters of CNN. It has been tested with the model containing the Titan data along with the classical CNN models, which are, namely, VGG16, ResNet50, etc.

To regularize the hyperparameters of SVM, the gradient descent method has come to focus to get the optimized value of a bilevel optimization model. This method has been validated by a few existing methods with the proposed strategy of SVM their model with few existing gradient descent methods and clarified that the proposed method for SVM is getting the better accuracy in terms of forecast and analysis [1].

The main contributions of this paper can be summarized as follows:

(1) We present a thorough cost-effective multiclass classification system that is capable of classifying brain tumor imaging into two different modern biomedical image datasets (Glioma Brain MRI Images).

(2) We develop a hybrid system that makes use of the Egyptian Vulture Optimization (EVO) algorithm along with the convolutional neural networks (CNN) to resolve complexity and increase the accuracy of CNN.

(3) We present broad investigational outcomes and comparative analysis with several AI-based models to provide more insights into the proposed hybrid system.

This paper has been organized as follows. Section 2 discusses the findings of other researchers in this area and preliminary remarks based on the related work. The research methodology adopted for model implementation and evaluation is discussed in Section 3. Section 4 provides a detailed discussion of the research layout and experimental setup. Sections 5 and 6 explain the evaluation metrics, the datasets features, the parameters setup, and the analysis of the results. Section 7 concludes this work with remarks on future work.

2. Related Work

Image classification is a recent development of the machine learning and deep learning era. Image classification and clustering techniques are used to solve the problems related to medical science [1], natural language processing, Haboush et al. [8] and Hammouri et al. [9], and other engineering fields [10]. In the field of medical science, identifying the brain tumor is a critical task and needs visual analysis, which can be solved by the image classification process. There were many traditional methods of image classification, and nowadays, CNN is a popular technique. In Rath et al. [1], the authors experimented with both the traditional methods and deep network techniques, and in their findings, they said that the CNN model performs with better accuracy than other models.

In Rath et al. [1], it was also found that the performance of CNN can be improved by hyperparameter tuning, which is a motivation for this work. Khairandish et al. [2] used a hybrid model of CNN and SVM to classify Benign and Malignant tumors from the brain MRI images. From the experiment, they found that the hybrid model of CNN and SVM performs with an accuracy 98.49%. In the same concern, Alsaffar et al. [11] evaluated several classification models including SVM, logistic regression, and nearest neighbors to analyze X-ray imaging to spot abnormalities. It was recorded that SVM achieved the best performance.

El-Dahshan et al. [3] used two hybrid classifiers, that is, feed-forward backpropagation artificial neural network (FP-ANN) and k-nearest neighbor (k-NN), to classify the MRI data, where they obtained an accuracy of 97% and 98% for FP-ANN and k-NN, respectively. In Jiang & Siddiqui [12], researchers tuned the parameters of support vector machines using stochastic gradient descent and dual coordinate descent to enhance the performance of SVM, which is a motivation for hyperparameter tuning of CNN.

Table 1: Summary of the state of art models.

Ref.	Model used	Advantage	Disadvantage
[1]	Hyper-tuned CNN using spotted hyena optimizer (SHO)	Improved accuracy 99.25%	Hyperparameter tuning model is considered complex
[2]	CNN and SVM	Accuracy 98.49%.	High computation complexity
[3]	Feedforward back-propagation artificial neural network (FP-ANN) and k-nearest neighbour (k-NN)	FP-ANN and k-NN are 97% and 98% respectively	No hyperparameters optimization
[14]	Hyper-tuned CNN using univariate dynamic encoding (UDE) algorithm.	Above 95% accuracy.	High implementation requirements
[15]	CNN optimized hyperparameters based on genetic algorithm	90% accuracy	Low accuracy and high computational complexity
[16]	Hyper-tuned parameters of (CNN) using multi-level particle swarm optimization (MPSO).	Above 95% accuracy	The complexity and performance depend on the dimensions of the search space.
[20]	Deep convolutional neural network (DCNN).	97.72% accuracy	High computation complexity
Wang et al. [18]	CNN with adaptive momentum optimization	95% accuracy	Low accuracy and high computational complexity

Yoo [13] focused on optimizing the hyperparameters of deep neural networks using dynamic encoding algorithm. The model performance was developed after hyperparameter tuning. Similarly, Aszemi and Dominic [14] describe how genetic algorithms optimize hypermeters to enhance the performance of CNN, while Cui and Bai [15] confirm that CNN https://www.sciencedirect.com/topics/engineering/convolutional-neural-networks performance is closely related to the efficient setup of model hyperparameters. Singh et al. [16] proposed a Multilevel Particle Swarm Optimization (MPSO) algorithm to find the architecture and hyperparameters of the CNN simultaneously.

In Musallam et al. [17], Deep Convolutional Neural Network (DCNN) architecture for effective diagnosis of Brian tumors is presented. The proposed model was considered lightweight and had a small number of max-pooling layers; convolutional and training iterations were conducted. The results reported indicated 97.72% detection accuracy. In the same concern, lightweight deep neural network model for image classification was presented in Wang et al. [18]; their aim was to improve the classification accuracy while reducing number of parameters. The proposed model was described as Dense-MobileNet, which utilizes the concept of dense blocks. The reported accuracy for Dense-MobileNet was 96.46%. An adaptive medical image classification based on CNN with adaptive momentum hyperparameter optimization was presented in Aytaç et al. [19]. This adaptive method has reduced classification error from 6.12 to 5.44%; hence, the detection accuracy was increased to 95% comparing to state-of-the-art CNN architectures while using the same datasets.

Table 1 below summarizes major related work being referred in this work with a focus towards its implemented model, overall achieved results, and limitations.

Previous research analysis has inspired this work to select a nature-inspired optimization algorithm to enhance the performance of CNN. Hence, the following preliminary remarks can be summarized:

(1) The biomedical images like CT scans or MRIs have a large quantity of data about the underlying tissue architecture [21]. Interpolation and interpretation of these numerous data are required to yield an exact evaluation and correct calculation of the diagnosis of any diseases.

(2) This type of process often requires machine learning techniques like classification in order to build an expert system working manually to ensure an uncommon and exact diagnosis of diseases, which is beneficial for the biomedical system.

(3) With the advent of different strategies developed for image classification, there are many tricks for analysis of the image. The biomedical images containing important information regarding different fields of the human body such as brain MRIs, detection of cancer or tumors, and reports of ultrasound are being analyzed by using machine learning techniques easily. For an accurate diagnosis, the biomedical images are being analyzed by the radiologist, and the clinical reports are being developed.

(4) The Computer-Aided Diagnosis (CAD) based methodologies are being developed and organized by various supervised or unsupervised machine learning techniques, which are applied to develop and organize the data with respect to the image classification. Here, we have studied the implementation of some traditional machine learning strategies along with some deep learning-based methodologies [22]. In this work, the focus is on choosing the best values for the hyperparameters in CNN to overcome the limitation of choosing the parameters manually.

(5) The recent works regarding the bioimage classification have been done excluding the validation of hyperparameters of CNN [23]. So, many metaheuristic optimization methods have gained importance to analyze the values of hyperparameters in tuned CNN. So, using EVO based optimization technique can give a better solution for this biomedical image classification.

3. Methodology

In this work, we have focused on the machine learning techniques along with the pedagogical method to get the optimized metaheuristic optimized solution. The biomedical image segregation based on issues and challenges is associated with the CAD-based systems for the classification of images and validating the key attributes of image classification. The input parameters of this work are as follows:

> Kaggle Repository is known as an authentic data repository to get the classifiers of the input parameters to collect the brain MRI images.
>
> The machine learning technique based on hyper-tuned parameters of EVO optimization technique and CNN have been combined to train the model to eradicate the limitations of the huge volume of data. The proposed model can be deployed for IoT health care diagnosis applications such as screening using real-time data [24].
>
> A direct mechanism for comparison has been made to classify the image-based of CNN and deep neural networks. The performance evaluation of the combined model is refined and checked for further improvements.

The authenticity of the proposed model has been validated by means of integrity, classification rate, precision rate, and accuracy. The error factor has been reduced during the pretraining and testing phase of the datasets. To check the validity of performance, the evaluation metrics graph has been designed. The experiment is implemented phase-wise.

4. Classifiers

The classifiers in machine learning scenario are a set of algorithms that automatically synchronize data into one or more than one set of classes. Machine learning algorithms are responsible for automating the tasks to be done manually to save large amount of time and energy to make process efficient and easier. The classifiers in machine learning scenario can be classified as traditional ANN-based classifiers, which are Multilayer Perceptron (MLP), Extreme-learning Machine (ELM) [1], and Ensemble learning [25]. In the case of ANN, the neural networks combine the behavior of different species behavior with machine learning tasks.

ANN comprises a large number of neurons that consist of input and output units. The task of the neuron is to receive the information from input units and process the operation over it in order to get the output and forward the result to its next unit. Mainly, ANN has two important variations, which are MLP and ELM. The deep learning classifiers are CNN and DNN. Recent research works are considering deep neural network classifiers in order to improve performance.

4.1. Multilayer Perceptron (MLP). MLP is considered as one of the easiest variations of ANN. It comprises one input layer and two output layers, which are the target output and output layer. MLP can function as an approximation function, which is used in various applications. The activation function sometimes dismisses the functional flow of network architecture due to mismanagement in the selection of appropriate interconnection weights.

4.2. Extreme Learning Machine (ELM). ELM is one of the traditional ANN network architectures, which is commonly known as feed-forward neural network. It works in two steps; firstly, the input layer is being initiated, and secondly the weights of the input layer and hidden layer are explored. The weights between the hidden layer and output layer are then calculated on the basis of Moore-Penrose Generalized Inverse method. In addition, ELM networks require least number of computations as compared to other existing methods in the less time-consuming scenario. It uses activation function without requiring a gradient descent method for calculating the values at the hidden layer. The choice of activation function affects the processing of determination network architecture. ELM networks can approximate any continuous nonlinear function $f\ (x)$ by satisfying the condition

$$\lim_{L=\infty}\sum_{i=1}^{L}\beta h_i(x)-f(x)=0. \tag{1}$$

The optimized value of the hidden layer is calculated by

$$\ddot{Y}=H\cdot\beta, \tag{2}$$

where $H\ (H_1, H_2, H_3, \ldots H_i)$ are the values of the hidden layer connected with the output layer, and β is the weight matrix of the hidden layer. The objective of ELM is to reduce the error in the model; therefore, the updated weights' matrix β can be calculated by the following equation:

$$\begin{gathered}+\cdot\ddot{Y},\\ \beta=H,\end{gathered} \tag{3}$$

where $+H$ is known as the Moore–Penrose generalized inverse of the hidden layer output.

In spite of the tremendous performance of ELM and its classification, it is hard to manage the issues of convolution related to image variations and object detection. The specific variation in exceeding the number of hidden nodes is the highest limitation of ELM. Calculation of output matrix is highly expensive.

4.3. Convolutional Neural Network (CNN). The other type of variation of ANN is the Convolutional Neural Network, which is based on the application area derived from the output generated from a series of actions. The structure of the network layer is derived from the series of actions from the processing of input to output, and the CNN architecture is illustrated in Figure 1. The CNN is influenced by the organization of connectivity patterns along with the visual cortex, which is quite matching with the neurons of brain MRI images [23]. Basically, the CNN networks are widely used for speech reorganization, audio-visual image classification, object identification, etc. Exploiting the

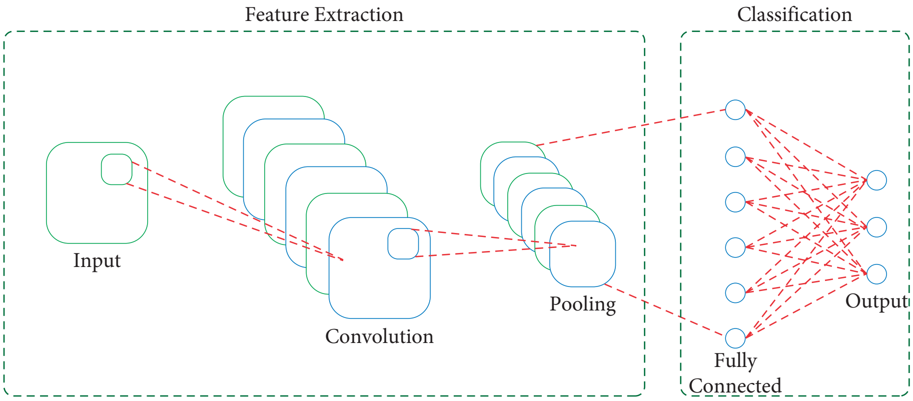


Figure 1: Basic architecture of the CNN model.

spatiotemporal behavior of input and output captured by the special layers of the CNN is beneficial for image classification. The general architecture of the classifier consists of convolution layers, input layers, and pooling layers. The CNN network takes an image as input and provides pixels as output [26]. The connected hidden layers are pooling layers with a lesser size to provide a large number of images to be trained.

A convolution tool that separates and identifies the various features of the image for analysis in a process is called Feature Extraction, while a fully connected layer utilizes the output from the convolution process and predicts the class of the image based on the features extracted in previous stages. Weight initialization is processed at the beginning using several techniques [27]. The main parts of the employed CNN architecture are as follows:

(a) Convolutional Layer: this layer is the first layer that is used to extract the various features from the input images. In this layer, the mathematical operation of convolution is performed between the input image and a filter of a particular size MxM. By sliding the filter over the input image, the dot product is taken between the filter and the parts of the input image with respect to the size of the filter (MxM). The output is termed the Feature map, which gives us information about the image such as the corners and edges. Later, this feature map is fed to other layers to learn several other features of the input image.

(b) Pooling layers: the primary aim of this layer is to decrease the size of the convolved feature map to reduce computational costs. This is performed by decreasing the connections between layers and independently operating on each feature map. Depending upon the method used, there are several types of pooling operations. In Max Pooling, the largest element is taken from the feature map. The Average Pooling calculates the average of the elements in a predefined sized Image section. The total sum of the elements in the predefined section is computed in Sum Pooling. The Pooling Layer usually serves as a bridge between the Convolutional Layer and the FC Layer

(c) Fully connected (FC) layers: the Fully Connected (FC) layer consists of the weights and biases along with the neurons and is used to connect the neurons between two different layers. These layers are usually placed before the output layer and form the last few layers of a CNN Architecture. In this, the input image from the previous layers is flattened and fed to the FC layer. The flattened vector then undergoes a few more FC layers where the mathematical function operations usually take place. In this stage, the classification process begins to take place.

(d) Dropout: usually, when all the features are connected to the FC layer, it can cause overfitting in the training dataset. Overfitting occurs when a particular model works so well on the training data causing a negative impact on the model's performance when used on new data. To overcome this problem, a dropout layer is utilized, wherein a few neurons are dropped from the neural network during the training process resulting in a reduced size of the model. On passing a dropout of 0.3, 30% of the nodes are dropped out randomly from the neural network.

(e) Activation Functions: finally, one of the most important parameters of the CNN model is the activation function. They are used to learn and approximate any kind of continuous and complex relationship between variables of the network. In simple words, it decides which information of the model should fire in the forward direction and which ones should not at the end of the network. It adds nonlinearity to the network. There are several commonly used activation functions such as the ReLU, Softmax, tanH, and the Sigmoid functions. Each of these functions has a specific usage. For a binary classification CNN model, sigmoid and Softmax functions are preferred for multiclass classification; generally, Softmax is used.

(f) The backpropagation algorithm along with the gradient descent search is one of the most used algorithms [28]. Different types of architecture of CNN networks are used in research and industrial fields in order to mitigate various limitations of ELM

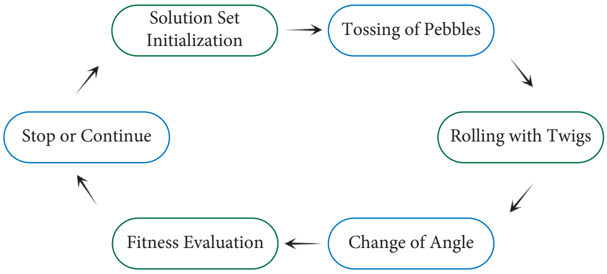


Figure 2: Steps for Egyptian vulture optimization algorithm [31].

algorithms. VGG-19, VGG-16, LeNet, and AlexNet are different types of CNN network architecture being used extensively nowadays.

4.4. Deep Neural Network (DNN). Different types of neural networks present in machine learning systems are one of the interesting key research areas for researchers nowadays. The simple type of neural network is a deep neural network, which is very much powerful and easy to implement [12]. It consists of two layers that are bounded with the high computational power of the devices of the network. The structure of the network is a helpful one in solving complex problems. The series of hidden layers is preceded by the single input layer, where the latter starts with the network architecture [13]. The last output layer is followed by the hidden layers, which are responsible to yield the output. The linear and nonlinear data transformations are carried out to give the output of the total process. The activation function is carried out at every hidden layer to perform the data transmission.

The backpropagation algorithm is one of the most common learning algorithms to train these kinds of neural networks [14]. Basically, the system architecture of DNN is responsible for the convergence of deep learning of the network with respect to nonlinear system processing. The selection of activation function to solve the problem is designed in such a way that the hyperparameters perform with the interconnection in the network [15]. A lot of optimization techniques have evolved in order to solve the deep learning problems with the association of tuned hyperparameters. DNN can be treated as a double-edged solution for interconnecting networks for obtaining the optimal value of these parameters [29]. The network design along with the activation function can highly affect the optimization results. The initial weighted values of the hyperparameters along with the convergence speed of the network can provide the best optimal solution for solving the problems. The output of the next layer could be calculated as follows using (4) and (5).

$$d^{n+1} = E\left(\sum_{i} X_{ij} + b_j^{n+1}\right), \tag{4}$$

$$X_{ij} = d_i^n w_{ij}^{(n,n+1)}, \tag{5}$$

X_{ij} is the input for the next layer; b is the bias, which is the connection between different layers.

CNN networks have a lot of variations that solve a large range of problems [16]. In the case of single-class and multiclass classification, the input and output of the matrix can be computed by the last layer of the output.

4.5. Egyptian Vulture Optimization (EVO). The EVO algorithm of the neural network is designed on the biological features and activities of Egyptian Vultures observed from different species [30]. The series of actions taken place by the searching and hunting of food as compared to other birds are quite unique. It is a metaheuristic optimization algorithm that was primarily initiated to solve difficult arrangement problems. It is arranged by the behavior of Egyptian Vulture to nourish its biological orders. The smooth functioning of this winged animal is transmitted to an efficient algorithm to solve a different kind of optimization problems. This kind of vulture is famous for its unique type of hunting capabilities. Raw flesh is the main food of these kinds of animals, but they find it comfortable to eat the eggs of different species. They use huge stones to break the hard surface of eggs. They use twigs to move fast for the sake of hunting and collecting food. These are two important features of this bird, which are remarkable activities and can be transformed into algorithms step by step. In Figure 2, the phases of EVO are depicted graphically. Also, the EVO Algorithm can be formulated as an algorithm as follows:

Step-1: The string of the solution set is first initialized with the presentation of parameters in the format of variables.

Step-2: The string displays the set of parameters that present a particular state of desirable solution.

Step-3: Then, the conditions are checked for a refined variable with the constraints. The constraints are superimposed.

Step-4: The stones that are used for breaking the eggs are selected randomly as a selection point.

Step-5: The selected parts of the entire string are then considered for twigs to be performed on a multiroller basis.

Step-6: Selected portion of the solution of the problem is then made reversed in order to change the tactic of the angle.

Step-7: The fitness value of the function is then evaluated.

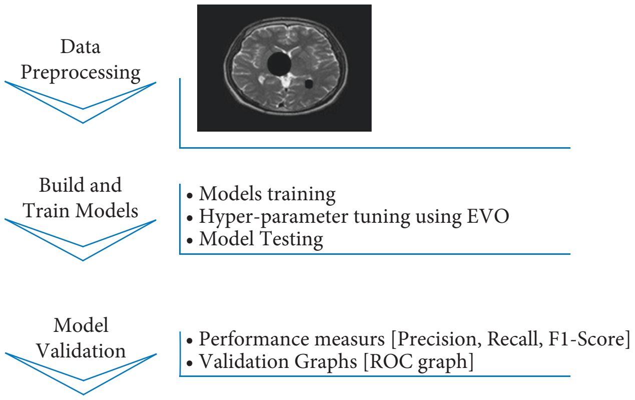

Data
Preprocessing
Build and
Train Models
• Models training
• Hyper-parameter tuning using EVO
• Model Testing
Model
Validation
• Performance measurs [Precision, Recall, F1-Score]
• Validation Graphs [ROC graph]

Figure 3: Overall design of the proposed biomedical image classification model.

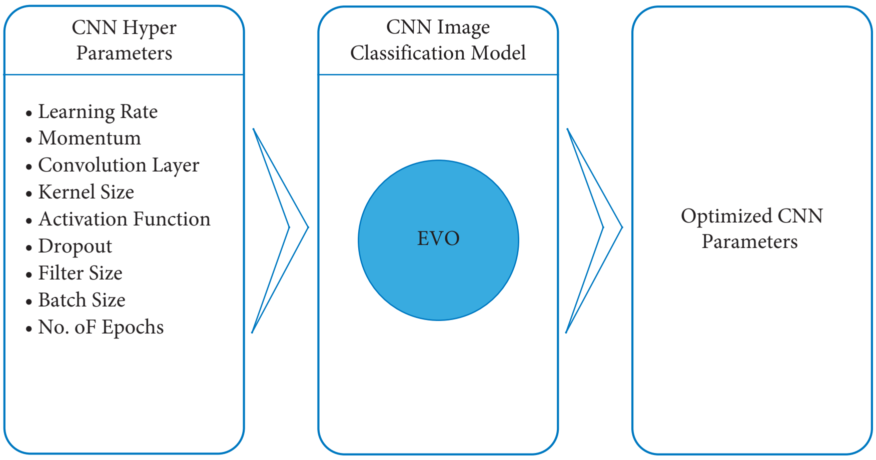


Figure 4: EVO tunes the hyperparameters of CNN.

Step-8: Then, checking the stooping criteria is done on regular basis.

4.5.1. Pebble Tossing. The EVO algorithm uses the biological features of the Egyptian Vulture, which breaks the hard eggs of different feathered birds to collect the food. After breaking the entire egg, the vulture can have their entire food. This method of action can build the technique for a route plan for action.

Determination of the operation is based on two variables known as Pebble Size (PS), which corresponds to the level of occupying, where PS is greater or equal to 0. The second variable is Force of Tossing (FT), which corresponds to the level of removal, and its value should be greater than or equal to zero. Accordingly, If $PS > 0$, then "Get In"; otherwise, "No Get In." In the same concern, if $FT > 0$, then "Removal"; otherwise, "No Removal." The level of occupying describes how many solutions they should pebble and carry. The level of removal describes how many solutions are taken out from the solution set. Hence, FT denotes the number of nodes removed [31]. Overall, there are four combinations of operations that are possible, which are as follows: Case 1: Get In and No Removal; Case 2: No Get In and No Removal; Case 3: Get In and Removal; Case 4: No Get In and No Removal [31].

4.5.2. Twigs Rolling. The activity of this type of vulture can find out the weakness of the twigs, which are moving in nature. After finding out the weakness of the twigs, the position of the object is moved towards the floor. The change in variable position of the activity can assume the modifications in the old position and can set new positions. The angel rotation and degree of modification affect the variables and parameters, which are being formulated mathematically to get the desired solution [32]:

(1) DS = Degree of roll

(2) DR as Direction of rolling

(3) $DR = 0$, if shift is right $= 1$, if shift is left

(4) DR = Left Rolling/Shift for RightHalf > LeftHalf = Right Rolling/Shift for RightHalf < LeftHalf

where 0 and 1 are generated randomly [6].

4.5.3. Changes in Angle. The vulture can rotate the angle of the stone to hit the egg. The rotation of angle depends on the analogy of the changes in angle with respect to pebble tossing in order to derive the experimental results of the process, which increases the chances of breaking the eggs.

Table 2: Description of dataset.

Dataset	Total data size	Classes and class labels	No. of samples used for training	No. of samples used for testing
Glioma	3264	Glioma tumor-0	826	100
		Pituitary-01	827	74
		Meningioma-02	822	115
		No tumor-03	395	105
Brain MRI images	253	No tumor-0	68	30
		Tumor- 1	108	47

Table 3: Parameters for MLP and ELM.

Layers	MLP	ELM
IL	NN = 22500	NN = 22500, AF = ReLU
HL	All the parameters are defined in Keras API	HL = 1, no. of nodes = 11250, AF = ReLU
OL	NN = 4	NN = 4, AF = Softmax

Table 4: Parameters for CNN.

Model	Sequential
CL 1	FS = 256, KS = (3, 3) Padding = same, AF = ReLu, IS = (75, 75, 1)
Max pool	PS = (3,3), dropout = 0.3
CL 2	FS = 128, KS = (3, 3) Padding = same, AF = ReLU
Max pool	PS = (2, 2), dropout = 0.3
CL 3	FS = 256, KS = (2, 2) Padding = same, AF = ReLU

Max pool	PS = (2, 2), dropout = 0.3
	Dense = 824, AF = ReLU

Optimizer	Adam, lr = 0.01, beta_1 = 0.7, beta_2 = 0.79
Epoch, batch size	Epoch = 50, batch size = 50

The more the chances are, the more it can break the harder eggs. The sequence of nodes can be completed by analyzing the sequence of the reverse of connected nodes. These changed angles are presented as mutation steps. The local search operation gives the decision of selecting the neighboring points. The connected graphs depend on the number of nodes of the string. Suppose that a string is full of nodes and can not handle Pebble tossing; then, Change in Angle is a good option for finding the solution [32].

5. Proposed Biomedical Image Classification Model

Biomedical image dataset is preprocessed and passes through different phases of classification. The flow diagram is explained in Figure 3. Brain tumor MRI images are collected from the repository; after that, the images are given as input to the CNN model. The CNN model process images as described in Section 4. The hyperparameters control the CNN model. The performance of the model depends on its hyperparameters. In this work, the EVO tunes the nine numbers of hyperparameters of CNN to get better accuracy. In Figure 4, it is depicted that the EVO tunes the CNN hyperparameters.

Table 5: Hyperparameters of CNN to be tuned.

Hyperparameters	Range
LR	Min. value-0.001, max. value-0.1
Mom.	Min. value-0.9, max. value-0.99
BS	Min. value-32, max. value-256
NN in CLs	Min. value-5, max. value-516
KS	[(2, 2), (3, 3), (5, 5), (6, 6)]
AFs	[ReLU, Sigmoid, LReLU]
FS	Min. value-64, max. value-256
Dropout	Min. value-0.1, max. value-0.9
No. Of epochs	Min. value-25, max. value-500

Table 6: Hyperparameters of CNN after tuning for the Glioma dataset.

Hyperparameters	Tuned value
LR	0.077
Mom	0.96
BS	64
No. of nodes in CL 2 and CL3	256
KS of IL, KS of CL 2, CL3	(3, 3)
AFs	ReLU
FS of IL, FS of CL 2, FS of CL3	128
The number of epochs	500

Table 7: Hyperparameters of CNN after tuning for Brain MRI dataset.

Hyperparameters	Tuned value
LR	0.005
Mom	0.95
BS	64
No. of nodes in CL 2 and CL3	128
KS of IL, KS of CL 2	(3, 3)
KS of CL 3	(5, 5)
AFs	ReLU
FS of IL, FS of CL 2 and CL3	64
The number of epochs	200

6. Research Metrics and Empirics

In this section, the system configuration, parameters, and datasets used are discussed for all the classification models, and also, the range of the hyperparameters for CNN is discussed.

6.1. System Configuration. The experimental evaluation is carried out in the Google Colab environment, under

Table 8: Performance of evaluation of traditional ANN-based strategies, DNN based strategies, and the proposed hyperparameter tuned CNN based on EVO for the Glioma dataset.

Models	Class	Accuracy	Precision	Recall	Specificity	F1-score	MCR	FDR	TNR
MLP	Glioma tumour	0.8372	0.6731	0.7000	0.8840	0.6863	0.1628	0.3269	0.3
	Meningioma tumour	0.8144	0.6612	0.7207	0.8520	0.6897	0.1856	0.3388	0.2793
	No tumour	0.8663	0.7609	0.7000	0.9239	0.7292	0.1337	0.2391	0.3
	Pituitary tumour	0.9049	0.7463	0.7143	0.9467	0.7299	0.0951	0.2537	0.2857
ELM	Glioma tumour	0.8878	0.7407	0.8333	0.9054	0.7843	0.1122	0.2593	0.1667
	Meningioma tumour	0.8817	0.7627	0.8333	0.9004	0.7965	0.1183	0.2373	0.1667
	No tumour	0.9302	0.8989	0.8163	0.9689	0.8556	0.0698	0.1011	0.1837
	Pituitary tumour	0.9308	0.8000	0.8116	0.9564	0.8058	0.0692	0.2	0.1884
DNN	Glioma tumour	0.9412	0.8700	0.8969	0.9558	0.8832	0.0588	0.13	0.1031
	Meningioma tumour	0.9487	0.9091	0.9091	0.9643	0.9091	0.0513	0.0909	0.0909
	No tumour	0.9523	0.9000	0.9000	0.9687	0.9000	0.0477	0.1	0.1
	Pituitary tumour	0.9512	0.8696	0.8571	0.9718	0.8633	0.0488	0.1304	0.1429
CNN	Glioma tumour	0.9820	0.9574	0.9677	0.9864	0.9626	0.018	0.0426	0.0323
	Meningioma tumour	0.9747	0.9402	0.9735	0.9752	0.9565	0.0253	0.0598	0.0265
	No tumour	0.9844	0.9890	0.9474	0.9966	0.9677	0.0156	0.011	0.0526
	Pituitary tumour	0.9746	0.9333	0.9333	0.9843	0.9333	0.0254	0.0667	0.0667
Hyperparameter tuned CNN	Glioma tumour	0.9847	0.9691	0.9691	0.9898	0.9691	0.0153	0.0309	0.0309
	Meningioma tumour	0.9847	0.9735	0.9735	0.9892	0.9735	0.0153	0.0265	0.0265
	No tumour	0.9923	0.9901	0.9804	0.9965	0.9852	0.0077	0.0099	0.0196
	Pituitary tumour	0.9873	0.9722	0.9589	0.9938	0.9655	0.0127	0.0278	0.0411

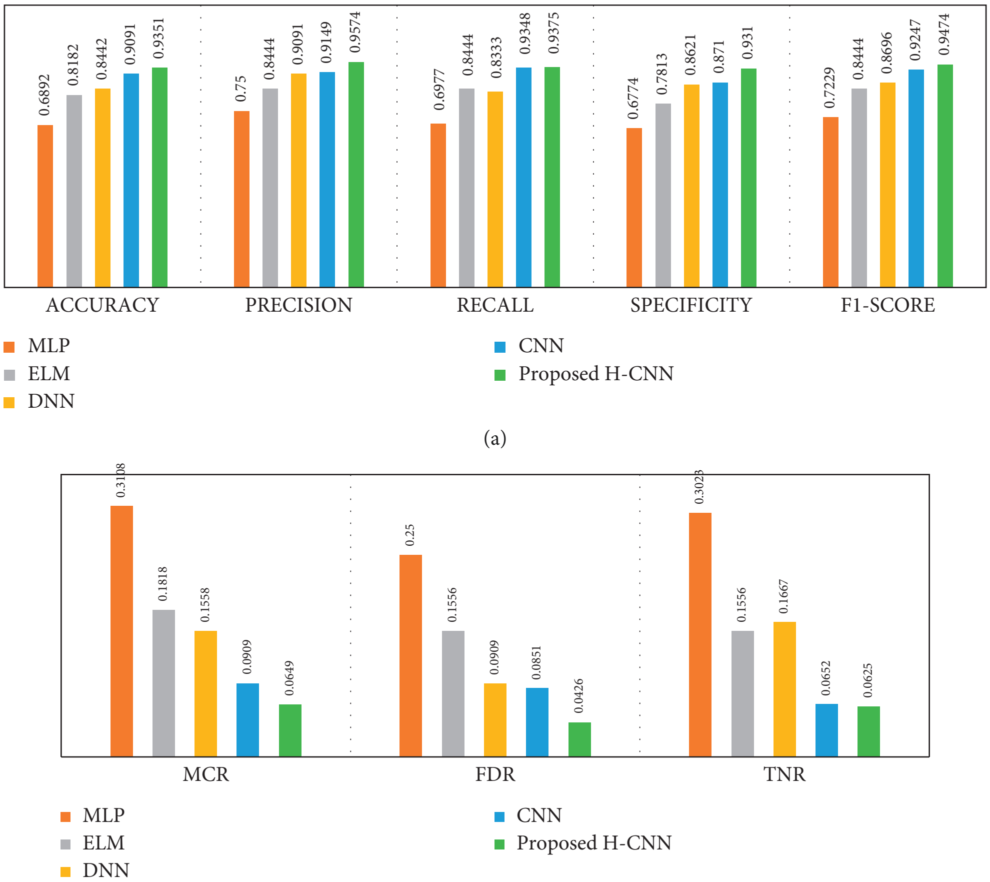
Windows7 with 64 bit and 4 GB RAM of Intel i3. Google Colab requires zero configurations with free access to GPUs to write and execute Python in the researcher's browser, which also provides an easy sharing facility. Additionally, this Colab harnesses the full power of popular libraries to analyze and visualize the data.

6.2. Datasets used for Experimentation. In this work, two datasets belonging to brain MRI are considered for experimental evaluation and comparison, which were collected from Kaggle Repository. The first dataset, known as "Glioma," has been published by Bhuvaji et al. [33], which is Brain MRI images. The second dataset being utilized is known as Brain MRI Images for Brain Tumor Detection, published and maintained by [34]. A detailed description of those two datasets is given in Table 2. In the Glioma dataset, 3264 samples are present, from which 826 Glioma tumor samples are used for training purposes, and 100 samples are used for testing purposes; similarly, 827 samples are used for training, and 74 samples, for testing for pituitary tumor class. From the Meningioma tumor class, 822 samples are used for training, and 115 samples, for testing, and from the no tumor class, 395 samples are used for training, and 105 samples, for testing purposes. In the second dataset, the total numbers of samples are 253, which contain two classes of data. The tumor present class contains 155 samples, out of which 108 samples are used for training purposes, and 47 samples are used for testing, and no tumor class contains 98 samples, out of which 68 samples are used for training purposes, and 30 samples are used for testing.

6.3. Parameters Selection. The CNN model has many predefined hyperparameters, which control the numbers of layers, nodes in each layer, etc. [35]. Similarly, in MLP, ELM, and DNN, the hyperparameters control the output of models [36–38]. As detailed in Table 3, the number of input nodes chosen for MLP and ELM is 2250. The number of nodes for the hidden layers for MLP utilized the Keras API; therefore, there is no need of choosing the number of hidden layers, number of nodes, and activation function. The ELM architectures use one hidden layer with 1300 nodes. The number of nodes in the output layer chosen for MLP and ELM is four. The ReLU activation function has been used for both input and hidden layers of ELM, whereas Softmax has been used for the output layer of ELM. Tables 4–7 represent CNN parameters, hyperparameters to be tuned, hyperparameters of CNN after tuning for Glioma dataset, and hyperparameters of CNN after tuning for the Brain MRI dataset.

7. Proposed Model Comparison and Validation

This section discusses the performance evaluation of the proposed optimized hyperparameters CNN based on the EVO algorithm. The proposed hyperparameter tuned CNN has been compared with variants of ANN-based and deep learning-based classifiers such as MLP, ELM, DNN, and CNN. The various performance measures based on classification accuracy have been recorded for both Glioma and Brain MRI image datasets. The result of hyperparameters values that minimize the cost as obtained by using the proposed biomedical image classifier is showing good results in comparison to other strategies considered for experimentation and comparison. Table 8 provides the detailed comparison results of MLP, ELM, DNN, and CNN over the proposed tuned model for every individual class. The comparison in the table considers the different output



(a)

(b)

Figure 5: Performance of evaluation of traditional ANN-based strategies, DNN-based strategies, and the proposed hyperparameter tuned CNN based on EVO for the Glioma dataset. (a) Quality factors, (b) Error rates.

Table 9: Observed training and testing average accuracy of hyperparameter tuned CNN for Glioma and Brain MRI datasets.

Datasets	Training (%)	Testing (%)
Glioma	97.34	98.72
Brain MRI images	90.98	93.51

classes including Glioma Tumor, Meningioma Tumor, No Tumor, and Pituitary Tumor. Also, the table compares the performance of the different Tumor classes in terms of classification accuracy, precision, recall, specificity, f1-score, misclassification rates (MCR), false discovery rate (FDR), and true negative rate (TNR).

Figure 5 provides the comparison results of MLP, ELM, DNN, and CNN over the proposed tuned model for the overall model tested over the Glioma dataset. The comparison considers two dimensions including: (A) quality factors, which compare the positive performance indicators such as accuracy, precision, recall, specificity, and F-score; (B) error rates, which compare the negative performance indicators such as MCR, FDR, and TNR. It can be clearly seen that hyperparameter tuned CNN outperforms other models scoring highest quality indicators and least error rates.

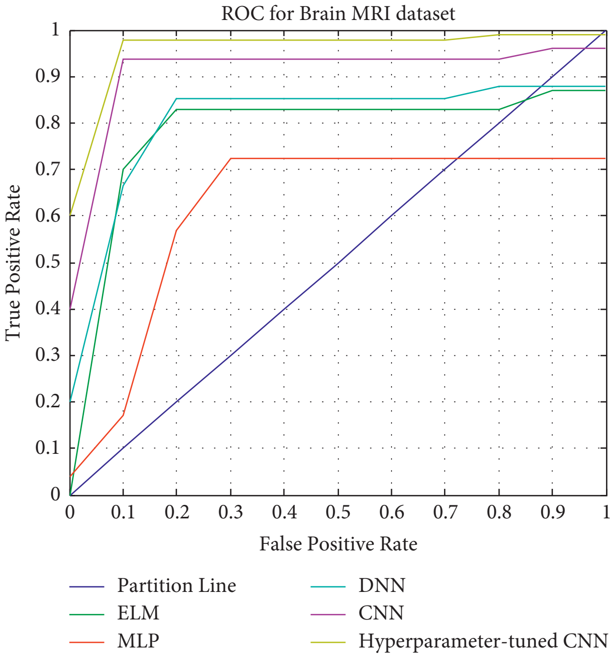


Figure 6: ROC for brain MRI dataset.

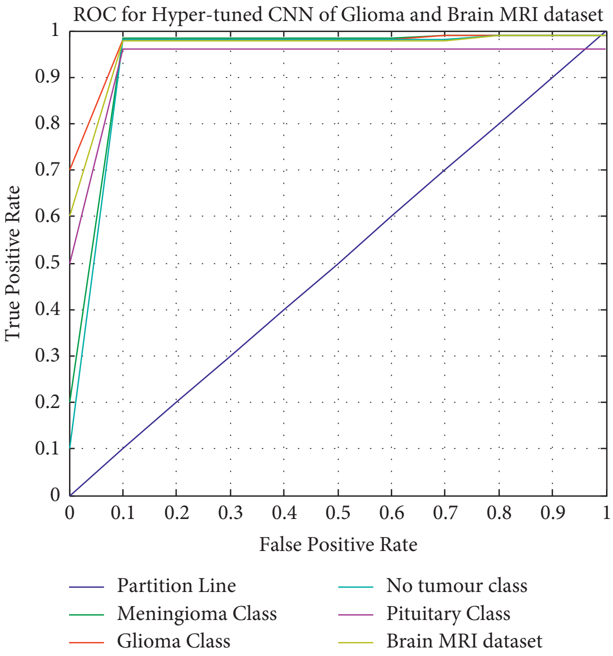


Figure 7: ROC for hyperparameter tuned CNN.

Table 9 reports the observed training and testing average accuracy of hyperparameter tuned CNN for Glioma and Brain MRI datasets. Consequently, the experimental simulations exhibit an improved classification performance for the hyperparameter tuned CNN when trained and tested using the Glioma dataset providing 98.7% of classification accuracy.

The area under ROC represents the accuracy of the classification models [39]. From Figures 6 and 7, it is observed that hyper-tuned CNN covers more area than other models. Therefore, it can be concluded that the hyper-tuned CNN performs better than the other experimented models.

8. Conclusion and Future Scope

This work presented an efficient classifier model focused on detecting cancer cell existence. After detecting a cancer cell, the second step is to find the category of cancer. This work proposed an optimized CNN based on the EVO approach, which has been compared with variants of ANN-based and deep learning-based classifiers such as MLP, ELM, DNN, and CNN. All model evaluation was based on two up-to-date datasets of Brian MRI images (Glioma and Brain MRI image datasets). The results achieved have confirmed the high performance of the CNN model after optimization and tuning of its hyperparameters based on the EVO technique. Results achieved by research lead towards investigating further optimization of hyperparameters of CNN and the integration of additional deep learning models for advanced detection performance. In the future, we will consider tuning our proposed model to perform high-performance classification tasks for other medical images such as lung cancer images and phasic dopamine releases [40]. Besides, the proposed system can be customized to provide detection in advance with high accuracy for several other health risks such as breast cancer detection [41], Tuberculosis Disease Diagnosis [42], and early-stage diabetes risk prediction [43]. Also, we will seek to develop a comparative study on the use of CNN with several other metaheuristic algorithms [20] such as particle swarm optimization (PSO) and Cuckoo Optimization Algorithm (COA) [42].

Data Availability

The [Glimo] data used to support the findings of this study have been deposited in the [Cancer Data Access System] repository ([https://cdas.cancer.gov/datasets/plco/16/]).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] P. Rath, P. K. Mallick, R. Siddavatam, and G. S. Chae, "An empirical development of hyper-tuned CNN using spotted hyena optimizer for bio-medical image classification," *Journal of Natural Science, Biology and Medicine*, vol. 12, no. 3, 2021.

[2] M. O. Khairandish, M. Sharma, V. Jain, J. M. Chatterjee, and N. Z. Jhanjhi, "A Hybrid CNN-SVM Threshold Segmentation Approach for Tumor Detection and Classification of MRI Brain Images," *IRBM*, Elsevier, Amsterdam, Netherland, 2021.

[3] E. S. A. El-Dahshan, T. Hosny, and A. B. M. Salem, "Hybrid intelligent techniques for MRI brain images classification," *Digital Signal Processing*, vol. 20, no. 2, pp. 433–441, 2010.

[4] M. K. Abd-Ellah, A. I. Awad, A. A. Khalaf, and H. F. Hamed, "A review on brain tumor diagnosis from MRI images: practical implications, key achievements, and lessons learned," *Magnetic Resonance Imaging*, vol. 61, pp. 300–318, 2019.

[5] G. Mohan and M. M. Subashini, "MRI based medical image analysis: survey on brain tumor grade classification," *Biomedical Signal Processing and Control*, vol. 39, pp. 139–161, 2018.

[6] S. Basheera and M. Satya Sai Ram, "A novel CNN based Alzheimer's disease classification using hybrid enhanced ICA segmented gray matter of MRI," *Computerized Medical Imaging and Graphics*, vol. 81, Article ID 101713, 2020.

[7] P. Patel, "Development of novel deep learning models in the detection of breast cancer and brain aneurysm," Santa Clara University, CA, USA, Bioengineering Master's Theses, 2021.

[8] A. Haboush, M. Al-Zoubi, A. Momani, and M. Tarazi, "Arabic text summarization model using clustering techniques," *World of Computer Science and Information Technology Journal (WCSIT) ISSN*, vol. 2, no. 3, pp. 2221–0741, 2012.

[9] A. Hammouri, M. Hammad, M. Alnabhan, and F. Alsarayrah, "Software bug prediction using machine learning approach," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 2, pp. 78–83, 2018.

[10] N. Panda and S. K. Majhi, "Effectiveness of swarm-based metaheuristic algorithm in data classification using pi-sigma higher order neural network," in *Progress in Advanced Computing and Intelligent Engineering*, pp. 77–88, Springer, Singapore, 2021.

[11] M. Alsaffar, G. Alshammari, A. Alshammari et al., "Detection of Tuberculosis disease using image processing technique," *Mobile Information Systems*, vol. 2021, Article ID 7424836, 7 pages, 2021.

[12] W. Jiang and S. Siddiqui, "Hyper-parameter optimization for support vector machines using stochastic gradient descent and dual coordinate descent," *EURO Journal on Computational Optimization*, vol. 8, no. 1, pp. 85–101, 2020.

[13] Y. Yoo, "Hyperparameter optimization of deep neural network using univariate dynamic encoding algorithm for searches," *Knowledge-Based Systems*, vol. 178, pp. 74–83, 2019.

[14] N. M. Aszemi and P. D. D. Dominic, "Hyperparameter optimization in convolutional neural network using genetic algorithms," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 6, pp. 269–278, 2019.

[15] H. Cui and J. Bai, "A new hyperparameters optimization method for convolutional neural networks," *Pattern Recognition Letters*, vol. 125, pp. 828–834, 2019.

[16] P. Singh, S. Chaudhury, and B. K. Panigrahi, "Hybrid MPSO-CNN: multi-level particle swarm optimized hyperparameters of convolutional neural network," *Swarm and Evolutionary Computation*, vol. 63, Article ID 100863, 2021.

[17] A. S. Musallam, A. S. Sherif, and M. K. Hussein, "A new convolutional neural network architecture for automatic detection of brain tumors in magnetic resonance imaging images," *IEEE Access*, vol. 10, pp. 2775–2782, 2022.

[18] W. Wang, Y. Li, T. Zou, X. Wang, J. You, and Y. Luo, "A novel image classification approach via dense-MobileNet models," *Mobile Information Systems*, vol. 2020, Article ID 7602384, 8 pages, 2020.

[19] U. C. Aytaç, A. Güneş, and N. A. Ajlouni, "A novel adaptive momentum method for medical image classification using convolutional neural network," *BMC Medical Imaging*, vol. 22, no. 1, 2022.

[20] N. A. Jebril and Q. Abu Al-Haija, "Cuckoo optimization algorithm (COA) for image processing," *Nature-Inspired Optimization Techniques for Image Processing Applications. Intelligent Systems Reference Library*, vol. 150, 2019.

[21] H. Shahamat and M. Saniee Abadeh, "Brain MRI analysis using a deep learning based evolutionary approach," *Neural Networks*, vol. 126, pp. 218–234, 2020.

[22] Y. Wang, H. Zhang, and G. Zhang, "cPSO-CNN: an efficient PSO-based algorithm for fine-tuning hyperparameters of convolutional neural networks," *Swarm and Evolutionary Computation*, vol. 49, pp. 114–123, 2019.

[23] S. Pan, H. Guan, Y. Chen et al., "Land-cover classification of multispectral LiDAR data using CNN with optimized hyper-parameters," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 166, pp. 241–254, 2020.

[24] H. Mukhtar, S. Rubaiee, M. Krichen, and R. Alroobaea, "An IoT framework for screening of COVID-19 using real-time data from wearable sensors," *International Journal of Environmental Research and Public Health*, vol. 18, no. 8, p. 4022, 2021.

[25] Q. Abu Al-Haija, M. Krichen, and W. Elhaija, "Machine-learning-based darknet traffic detection system for IoT applications," *Electronics*, vol. 11, no. 4, p. 556, 2022.

[26] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, 2020.

[27] W. Boulila, M. Driss, E. Alshanqiti, M. Al-Sarem, F. Saeed, and M. Krichen, "Weight initialization techniques for deep learning algorithms in remote sensing: recent trends and future perspectives," *Advances on Smart and Soft Computing*, vol. 1399, pp. 477–484, 2022.

[28] P. Kumar, G. G. Nair, and G. G. Nair, "An efficient classification framework for breast cancer using hyper parameter tuned Random Decision Forest Classifier and Bayesian Optimization," *Biomedical Signal Processing and Control*, vol. 68, Article ID 102682, 2021.

[29] J. Kang, Z. Ullah, and J. Gwak, "MRI-based brain tumor classification using Ensemble of deep features and machine learning classifiers," *Sensors*, vol. 21, no. 6, p. 2222, 2021.

[30] C. Sur, S. Sharma, and A. Shukla, "Egyptian vulture optimization algorithm–a new nature inspired meta-heuristics for knapsack problem," in *Proceedings of the 9th International Conference on Computing and InformationTechnology (IC2IT2013)*, pp. 227–237, Springer, Bangkok, Thailand, May 2013.

[31] K. Dilip and M. Nandhini, "Adapting Egyptian vulture optimization algorithm for vehicle routing problem," *International Journal of Computer Science and Information Technologies*, vol. 7, no. 3, 2016.

[32] A. S. Tayeb and H. Bouzeboudja, "Application of a new meta-heuristic algorithm using egyptian vulture optimization for economic," *Przeglad Elektrotechniczny*, vol. 1, no. 6, pp. 58–67, 2019.

[33] S. Bhuvaji, A. Kadam, P. Bhumkar, S. Dedge, and S. Kanchan, "Brain tumor classification (MRI): classify MRI images into four classes," *Kaggle*, vol. 45, 2020.

[34] N. Chakrabarty, "Brain MRI images for brain tumor detection," 2019, https://www.kaggle.com/datasets/navoneel/brain-mri-images-for-brain-tumor-detection.

[35] Y. Cai, Z. Zhang, Q. Yan, D. Zhang, and M. J. Banu, "Densely connected convolutional extreme learning machine for hyperspectral image classification," *Neurocomputing*, vol. 434, pp. 21–32, 2021.

[36] G. Yang and F. Ding, "Associative memory optimized method on deep neural networks for image classification," *Information Sciences*, vol. 533, pp. 108–119, 2020.

[37] N. Ghassemi, A. Shoeibi, and M. Rouhani, "Deep neural network with generative adversarial networks pre-training for brain tumor classification based on MR images," *Biomedical Signal Processing and Control*, vol. 57, Article ID 101678, 2020.

[38] M. V. Rajee and C. Mythili, "Gender classification on digital dental x-ray images using deep convolutional neural network," *Biomedical Signal Processing and Control*, vol. 69, Article ID 102939, 2021.

[39] W. A. Yousef, R. F. Wagner, and M. H. Loew, "Comparison of non-parametric methods for assessing classifier performance in terms of ROC parameters," in *Proceedings of the 33rd Applied Imagery Pattern Recognition Workshop (AIPR'04)*, pp. 190–195, IEEE, Washington, DC, USA, October 2004.

[40] Q. A. Al-Haija, M. Smadi, and O. M. Al-Bataineh, "Identifying Phasic Dopamine Releases Using DarkNet-19 Convolutional Neural Network," in *Proceedings of the IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2021*, pp. 1–5, Toronto, Canada, April 2021.

[41] Q. A. Al-Haija and A. Adebanjo, "Breast Cancer Diagnosis in Histopathological Images Using ResNet-50 Convolutional Neural Network," in *Proceedings of the 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, pp. 1–7, Vancouver, British Columbia, Canada, September 2020.

[42] O. Hrizi, K. Gasmi, I. Ben Ltaifa, H. Alshammari, H. Karamti, and M. Krichen, "Tuberculosis disease diagnosis based on an optimized machine learning model," *Journal of Healthcare Engineering*, vol. 2022, Article ID 8950243, 13 pages, 2022.

[43] Q. A. Al-Haija, M. Smadi, and O. M. Al-Bataineh, "Early stage diabetes risk prediction via machine learning," in *Proceedings of the 13th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2021). SoCPaR 2021. Lecture Notes in Networks and Systems*, Springer, Seattle, WA, USA, December 2021.