


Research Article

An Efficient Method for Diagnosing Brain Tumors Based on MRI Images Using Deep Convolutional Neural Networks

Thanh Han-Trong ¹, **Hinh Nguyen Van**,¹ **Huong Nguyen Thi Thanh**,¹ **Vu Tran Anh**,¹
Dung Nguyen Tuan,² and **Luu Vu Dang**²

¹*School of Electrical and Electronic Engineering, Hanoi University of Science and Technology, Hanoi 100000, Vietnam*

²*Bach Mai Hospital, Hanoi 100000, Vietnam*

Correspondence should be addressed to Thanh Han-Trong; thanh.hantrong@set.hust.edu.vn

Received 19 November 2021; Revised 22 January 2022; Accepted 23 February 2022; Published 15 March 2022

Academic Editor: Abidhan Bardhan

Copyright © 2022 Thanh Han-Trong 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 paper proposes a system to effectively identify brain tumors on MRI images using artificial intelligence algorithms and ADAS optimization function. This system is developed with the aim of assisting doctors in diagnosing one of the most dangerous diseases for humans. The data used in the study is patient image data collected from Bach Mai Hospital, Vietnam. The proposed approach includes two main steps. First, we propose the normalization method for brain MRI images to remove unnecessary components without affecting their information content. In the next step, Deep Convolutional Neural Networks are used and then we propose to apply ADAS optimization function to build predictive models based on that normalized dataset. From there, the results will be compared to choose the most optimal method. Those results of the evaluated algorithms through the coefficient F1-score are greater than 94% and the highest value is 97.65%.

1. Introduction

The brain is a particularly important organ, the control center of the central nervous system, coordinating the activities of all organs and parts in the human body. The brain has a complex structure and is protected and covered by the skull, a very hard bone box. However, a rigid skull may help protect the brain parenchyma from minor trauma but does not prevent the development of lesions and abnormal structures within the brain. One of the brain diseases of primary concern in medicine is brain tumors. A brain tumor is a condition in which abnormal cells grow in the brain. Brain tumors are divided into two types: benign brain tumors and malignant brain tumors (called cancer) [1]. Whether it is a benign brain tumor or a malignant brain tumor, it affects brain cells, causing brain damage and being even life-threatening. There are about 120 different types of brain tumors, most of which are tumors in the brain tissue, in addition to tumors in the meninges, pituitary gland, cranial nerves. Any form of brain tumor can be dangerous

for the patient. Tumors in brain tissue or benign brain tumors often progress slowly; the symptoms of brain tumors in this case will also appear slower and more insidious. In contrast, if the brain tumor grows rapidly, the patient will feel the symptoms more pronounced in both frequency and extent. With current medical capabilities, early detection of abnormal structures in the patient's brain can improve the likelihood of successful treatment and limit the sequelae of tumors to the brain in general and the patient's health in particular.

The detection of brain tumors today is mostly based on the ability of doctors to distinguish abnormalities on MRI images which is a type of high-quality image in the field of imaging [2]. This is a process that requires a lot of experience and concentration to detect and classify brain diseases and brain tumors. From brain MRI, it is possible to diagnose and recognize many different types of brain tumors and offer appropriate treatment methods [3]. However, the increasing number of patients with the large number of images obtained has become a major challenge in the field of

diagnostic imaging, a field that requires rapid and accurate evaluation of results by doctor. Artificial intelligence technology will help classify diseases from MRI images quickly and bring high accuracy in disease diagnosis. The classification of diseases based on MRI images has not been too difficult with high accuracy due to the introduction of GPUs (Graphics Processing Unit) and image processing based on artificial intelligence (AI).

This research focuses on the application of image pre-processing techniques and the development of algorithms using convolutional neural network (CNN) models, which are advanced deep learning models such as DenseNet201 [4], ResNet152V2 [5], MobileNetV3 [6], and VGG19 [7]. At the same time, the research also focuses on developing and applying the ADAS optimization algorithm to improve the accuracy in classifying normal people and brain tumor patients. The dataset in this work includes 1307 brain MRI images in JPEG format that are manually classified by specialists into 2 categories: normal human brain MRI images and brain MRI images of people with brain tumor disease. The comparison of all experimental results will evaluate the effectiveness of each model.

The article is organized as follows. Section 2 presents the previously conducted MRI brain tumor classification studies. Section 3 provides an overview of brain MRI images and the CNN algorithm models used. Section 4 presents the experimental results and gives evaluation for each algorithm. Conclusions and future work are outlined in Section 5.

2. Related Work

Several technical methods related to brain MRI images classification since 2017 based on different classification models are summarized in Table 1. They are divided into two basic methods: using CNN network architecture and not using CNN network architecture. In [10], the authors divided brain MRI images into two categories: normal images and images with abnormal signs. They used GLCM to get the features of the MRI images; then a probabilistic neural network (PNN) was used to classify the MRI brain images of people as normal or abnormal. As a result, they obtained a classification model with an accuracy of 95%. In [14], Ullah et al. proposed a scheme to classify the brain MRI images of normal people and patients using equilibrium histograms, discrete wavelet transforms, and Feedforward Artificial Neural Networks. Recently, deep learning method has been widely used for the classification of brain tumors on MRI images [8, 9]. The deep learning method does not need to manually extract the features of the images; it combines the extraction and classification stages in the self-learning process. The deep learning method requires a dataset where normalized processing of the MRI images is sometimes required, and then salient features are identified during machine learning [13].

Convolution Neural Network (CNN), one of the well-known deep learning techniques for image data, can be used as a feature extraction tool from which to capture related features to perform data classification task. Feature maps in the initial and higher layers of the CNN model extract

low-level features and specific features of high-level content, respectively. Feature maps in the earlier layer construct simple structural information, such as shapes, textures, and edges, while the higher layers combine these low-level features into constructing (encoding) expressions performance, integrating local and global information.

Various researchers have proposed to use CNN to classify brain tumors based on brain MRI image datasets [11, 21, 22]. Deepak and Ameer [12] used pretrained GoogLeNet to extract features from brain MRI images with CNN network architecture to classify three types of brain tumors and obtained up to 98% accuracy. Çınar and Yildirim, [15] modified the ResNet50 network based on the pretrained CNN network architecture by removing the last 5 layers and adding 8 new layers, and that method achieved 97.2% accuracy. Saxena et al. [17] used InceptionV3, ResNet50, and VGG16 network architectures with legacy methods to classify brain tumor data. In this study, ResNet50 model obtained the highest accuracy rate with 95%. Díaz-Pernas et al. [18] presented a CNN network architecture for automatic brain tumor segmentation such as glioma, meningioma, and pituitary tumor. They evaluated their proposed model using the T1-weighted contrast-enhanced MRI dataset and obtained an accuracy of 97.3%.

Siddiaue et al. [16] proposed a model based on modified vgg-16 network architecture for brain tumor images classification which achieved an accuracy of 96% and an F1-score of 97%. Abd El Kader et al. [19] developed a differential deep-CNN-based model to classify MRI images with and without tumors. In fact, this model was still based on the basic CNN architecture but obtained an accuracy of 99.25% and an F1-score of 95.23%. In [20], the authors successfully deployed transfer learning for some variant architectures of CNN to apply to the classification of MRI images with and without brain tumors, in which MobileNetV2 had an accuracy of 92% and F1-score of 92%; InceptionV3 had an accuracy of 91% and an F1-score of 90.98%; VGG19 had an accuracy of 88% and F1-score of 88.18%.

In summary, as observed from the above studies, the accuracy obtained by using deep learning with CNN network architecture to classify brain MRI is significantly higher than that of the old traditional techniques. However, deep learning models require a large amount of data to train in order to perform better than traditional machine learning techniques.

3. Materials and Methods

3.1. Brain Tumor MRI Images

3.1.1. Content Contained in MRI Images. The commonly used standard for MRI images today is DICOM, an acronym for Digital Imaging and Communications in Medicine Standards [23]. This is an industry standard system developed to meet the needs of manufacturers and users in connecting, storing, exchanging, and printing medical images.

As for the DICOM image format standard, in addition to the image files, it also includes header files as in Figure 1.

TABLE 1: Summary of studies on brain tumor classification.

Author	Classification method	Objective	Dataset	Feature extraction method	Accuracy
Khawaldeh et al. [8]	CNN	Classification of brain MRI into normal and abnormal	587 MR images	CNN	91.16%
Paul et al. [9]	Fully connected and CNN	Brain tumor classification of MR brain image	3064 MR images	CNN	91.43%
Varuna Shree and Kumar [10]	Probabilistic neural network (PNN)	Classification of brain MRI into normal and abnormal	650 MR images	Gray level cooccurrence matrix (GLCM)	95%
Hemanth et al. [11]	CNN	Classification into normal and abnormal	220 MR images	CNN	94.5%
Deepak and Ameer [12]	Deep transfer learning	Classification of glioma, meningioma, and pituitary tumors	3064 MR images	Google Net	98%
Das et al. [13]	CNN	Brain tumor classification	3064 MR images	CNN	94.39%
Ullah et al. [14]	Feedforward neural network	Classification of brain MRI into normal and abnormal	71 MR images	DWT	95.8%
Çinar and Yildirim [15]	CNN models	Brain tumor detection and classification	253 MR images	CNN	97.2%
Siddiaue et al. [16]	Proposed DCNN model	Brain tumor classification	253 MR images	CNN	96%
Saxena et al. [17]	CNN networks with transfer learning	Binary classification of brain tumor normal and abnormal	253 MR images	CNN	95%
Díaz-Pernas et al. [18]	Multipathway CNN	Brain tumor classification	3064 MR images	CNN	97.3%
Abd El kader et al. [19]	Proposed differential deep-CNN	Brain tumor classification	25000 MR images	CNN	99.25%
Tazin et al. [20]	CNN architectures	Brain tumor classification	2513 MR images	CNN	Up to 92%

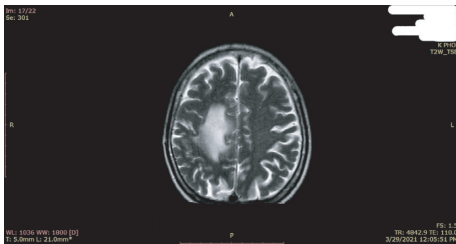


FIGURE 1: Actual DICOM image (patient information has been removed).

Although stored in different files, when displayed, the header information is displayed along with the MRI image information via a “DICOM browser.” Data in MRI images include demographic information, patient information, parameters acquired for imaging studies, image size, and image matrix size. The patient’s information displayed includes patient’s first and last name, gender, age, date of birth, and place where the MRI scan was performed.

3.1.2. The Role of MRI Images in the Diagnosis of Brain Tumors. Magnetic resonance imaging of the brain [24] can very clearly detect and describe abnormalities in the brain parenchyma in general such as vascular tumors, arterial occlusion, and invasion of the venous sinuses as well as the relationship between tumor and surrounding structures. There are three basic image formats of MRI images: T1W, T2W, and T2 Flair. They are used in specific cases depending on the situation of the disease.

T1W imaging is mainly used to identify necrotic tumors, hemorrhage in tumors, or cysts. For example, with MRI images in meningiomas, on T1-weighted images, most meningioma shows no difference in signal intensity compared with cortical gray matter.

For image in the T2W phase, the received signal has been changed completely; it is a fairly homogeneous gain signal block. Imaging is also helpful in evaluating hemorrhages and cysts. In particular, the role of the T2W phase is very useful in reflecting the homogeneity of benign soft tumors or meningiomas.

For Fluid-Attenuated Inversion Recovery (T2-FLAIR), this type of phase image is very useful to evaluate the consequences and effects of edema. Although this finding is not specific for meningiomas in particular, it is very meaningful in the diagnosis as well as the long-term prognosis for the patient.

Overall, the sensitivity and specificity of MRI are very high in the diagnosis of meningiomas. MRI has been shown to be superior in tumor delineation by its relationship to surrounding structures.

3.2. Model Architectures

3.2.1. Supervised Learning. Supervised learning [25] is an algorithm that predicts the output (outcome) of a new data (new input) based on previously known (input, outcome) pairs. This data pair is also known as (data, label). Supervised learning is the most popular group of machine learning algorithms.

Mathematically, supervised learning consists of a set of input variables $X = \{x_1, x_2, \dots, x_N\}$ and a corresponding set of labels $Y = \{y_1, y_2, \dots, y_N\}$, where x_i and y_i are vector. The data pairs $(x_i, y_i) \in X \times Y$ are called the training dataset. From this training dataset, we need to create a function that maps each element from the set X to a corresponding (approximate) element of the set Y as

$$y_i \approx f(x_i), \quad \forall i = 1, 2, \dots, N. \quad (1)$$

The goal is to approximate the function f very well so that when we have a new data x we can compute its corresponding label:

$$y = f(x). \quad (2)$$

A problem is called classification if the labels of the input data are divided into a finite number of groups.

3.2.2. Convolutional Neural Network Architectures. Convolutional Neural Network (CNN) [31] is one of the most popular and most influential deep learning models in the computer vision community. CNN is used in many problems such as image recognition and video analysis or for problems in the field of natural language processing and solves most of these problems well.

CNN includes a set of basic layers such as convolution layer, nonlinear layer, pooling layer, and fully connected layer. These layers are linked together in a certain order. Basically, an image will be passed through the convolution layer and nonlinear layer first; then the calculated values will be passed through the pooling layer to reduce the number of operations while preserving the characteristics of the data. The convolution layer, nonlinear layer, and pooling layer can appear one or more times in the CNN network. Finally, the data is passed through fully connected network and soft-max to calculate the probability of object classification.

Table 2 summarizes some typical CNN network architectures since 2012. To evaluate and compare network structures, two parameters are used, Top 1 Accuracy and Top 5 Accuracy. In the case of Top 1 Accuracy, the correct model's prediction must be the model that predicts the class with the highest probability. In the case of Top 5 Accuracy, the correct model's prediction is the model that correctly predicts one of the 5 classes with the highest probability.

In this study, four different network architectures are used: DenseNet201 [4], ResNet152V2 [5], MobileNetV3 [6], and VGG19 [7]. All the above four network architectures are developments and upgrades based on the basic network architecture CNN, one of the advanced deep learning models for image classification that has been verified with high accuracy on image sets, ImageNet [32]. These CNN variant network architectures are widely used in image recognition and classification problems. All four network architectures have a structure consisting of two basic layers, the feature extraction layer and the classifier layer. In this research, the input to the network architecture is a 256×256 brain MRI image containing information with or without brain tumors. The feature extraction layer has the role of

extracting features of brain MRI images such as white matter, gray matter, cerebrospinal fluid, cerebral cortex, and brain tumor. Then, the classification layer is responsible for synthesizing the features of brain MRI images, giving specific features of images with tumors and images without tumors to serve the classification process.

3.3. Optimal Algorithms. The optimization algorithm is the basis for building a neural network model with the aim of "learning" the features (or patterns) of the input data, from which it is possible to find a suitable pair of weights and biases to optimize the model. But the question is how to "learn?" Specifically, how the weights and biases are found, not just randomly taking the weights and biases values for a finite number of times and hoping after some steps a solution can be found. Therefore, it is necessary to find an algorithm to improve weights and biases step by step, and that is why optimizer algorithms were created.

Some of the factors commonly used to evaluate an optimizer algorithm are as follows:

- Fast convergence (Training Process)
- High generalization (can recognize previously untrained patterns)
- High accuracy

The popular optimization algorithms are listed in Figure 2. These algorithms are GD-Gradient Descent [33], SGD-Stochastic Gradient Descent [34], ADAS-Adaptive Scheduling of Stochastic Gradients [35], AdaGrad-Adaptive Subgradient Methods for Online Learning and Stochastic Optimization [36], Momentum [37], RMSProp [33], and ADAM-Adaptive Moment Estimation [38].

Among the above algorithms, the Optimal Algorithms belonging to the adaptive family usually have fast convergence speed. Meanwhile, algorithms belonging to the SGD family often have high generalization. However, this study only focuses on the development and application of ADAM and ADAS algorithms.

3.3.1. ADAM Algorithm: A Method for Stochastic Optimization. ADAM is a combination of Momentum and RMSProp. One of the key components of ADAM is exponential weighted moving averages (also known as leak averages) that estimate both the momentum and the second-order moment of the gradient. Specifically, it uses state variables as follows:

$$\begin{aligned} v &\leftarrow \beta_1 v_t - 1 + (1 - \beta_1) g_t, \\ s &\leftarrow \beta_2 s_t - 1 + (1 - \beta_2) g_t^2, \end{aligned} \quad (3)$$

where v is the first moment vector, s is the second moment vector, β_1 and β_2 are the jump parameters at the initial and the second points in ADAM's algorithm, t is the time for the correction steps, and g is the gradient. Here β_1 and β_2 are nonnegative weight parameters. Popular choices for them are $\beta_1 = 0.9$ and $\beta_2 = 0.999$. This means that the variance estimate moves much slower than the momentum term.

TABLE 2: Some CNN architectures and ImageNet benchmark (image classification).

CNN architectures	Author	Top 1 accuracy (%)	Top 5 accuracy (%)
AlexNet	Krizhevsky et al. [26]	63.3	84.6
VGG16	Simonyan and Zisserman [7]	74.4	91.9
VGG19		74.5	92.0
GoogLeNet (InceptionV1)	Szegedy et al. [27]	69.8	89.5
ResNet50	He et al. [5]	77.1	93.3
ResNet152		78.6	94.3
DenseNet201	Huang et al. [4]	77.4	93.7
MobileNet224	Howard et al. [6]	70.6	89.5
MobileNetV3		79.0	94.5
ResNeXt101	Xie et al. [28]	80.9	95.6
EfficientNet-L2	Tan and Le [29]	90.2	98.8
RegNet-Y	Radosavovic et al. [30]	79.9	95.0

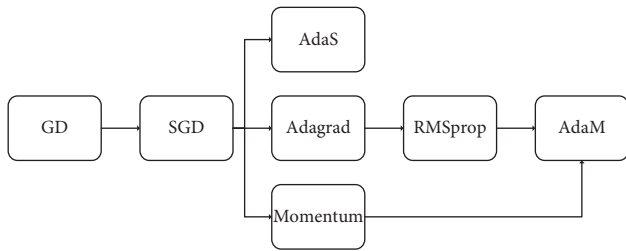


FIGURE 2: The development of optimization algorithms.

Note that if initializing the values $v_0 = s_0 = 0$, the algorithm will have a significant initial bias towards smaller values. This problem can be solved using $\sum_{i=0}^t (1 - \beta_i) / (1 - \beta)$ to normalize the terms. Similarly, state variables are normalized as follows:

$$v'_t = \frac{v_t}{1 - \beta_1^t}, \quad (4)$$

$$s'_t = \frac{s_t}{1 - \beta_2^t}.$$

From the appropriate estimates, the updated equations can be established. First, the gradient value will be adjusted, similar to that in RMSProp [33] to get

$$g'_t = \frac{\eta s'_t}{\sqrt{s'_t + \varepsilon}}, \quad (5)$$

where ε is a constant and it is chosen to be $\varepsilon = 10^{-6}$ to balance arithmetic stability and reliability, and η is the learning rate. From there, the update step is defined as follows:

$$x_t \leftarrow x_t - 1 - g'_t. \quad (6)$$

When looking at the design of ADAM, the inspiration of the algorithm is clear. Momentum and range are clearly represented in the state variables. Moreover, based on RMSProp it is easy to see that the combination of both terms is quite simple. Finally, the learning rate η allows us to control the update step length to solve convergence problems.

3.3.2. ADAS Algorithm: Adaptive Scheduling of Stochastic Gradients. ADAS [35] is an optimization algorithm belonging to the family of Stochastic Gradient Descent (SGD) algorithms. The updated rules for ADAS are established using SGD with momentum as follows:

$$\eta(t, l) \leftarrow \beta \cdot \eta(t-1, l) + \zeta \cdot [\bar{G}(t, l) - \bar{G}(t-1, l)],$$

$$v_l^k \leftarrow \bar{G} \alpha \cdot v_l^{k-1} - \eta(t, l) \cdot g_l^k, \quad (7)$$

$$\theta_l^k \leftarrow \bar{G} \theta_l^{k-1} + v_l^k,$$

where η is the learning rate, t is the time for the correction steps, β is ADAS gain factor, ζ is the knowledge gain hyperparameter, k is the current minibatch, t is the current epoch iteration, l is the convolution block index, $\bar{G}(\cdot)$ is the average knowledge gain obtained from both mode-3 and mode-4 decompositions, v is the velocity term, and θ is the learnable parameter. The learning rate is calculated relative to the rate of change of knowledge acquired after the training epochs. The learning rate $\eta(t, l)$ is then further updated by an exponential moving average called the gain factor, with the hyperparameter β , to accumulate the history of the knowledge gained over the series epochs. In fact, β controls the trade-off between convergence rate and training accuracy of ADAS.

ADAS is an adaptive optimization tool for scheduling the learning rate in the training of a CNN network. ADAS exhibits a much faster convergence speed than other optimization algorithms. ADAS demonstrated generalization characteristics (low test loss) on par with SGD-based optimizers, improving on the poor generalization characteristics of adaptive optimizers. In addition to optimization, ADAS introduces new polling metrics for CNN layer removal (quality metrics).

3.4. Accuracy and F1-Score. The classification problem in this study is a binary classification problem, in which one class is an MRI image with a brain tumor and the other is an MRI image without a brain tumor. This study considers the image class with brain tumor to be positive and the remaining image class without brain tumor to be negative. The parameters True Positive (TP), False Positive (FP), True

Negative (TN), and False Negative (FN) are described as in Table 3.

In this paper, the parameters used to evaluate the effectiveness of the model are accuracy, precision, recall, and F1-score [39]. When building a classification model, the ratio of correctly predicted cases to the total number of cases is always considered. That ratio is called accuracy. Precision is the answer to the question: how many true positives are there out of the total number of positive diagnoses? Recall measures the rate of correctly predicting positive cases across all samples in the positive group. F1-score is the harmonic mean between precision and recall. Therefore, in situations where the precision and recall are too different, the F1-score will balance both values and help us to make an objective assessment. Accuracy, precision, recall, and F1-score are defined as the following equations:

$$\begin{aligned} \text{accuracy} &= \frac{\text{TP} + \text{TN}}{\text{total sample}} \\ \text{precision} &= \frac{\text{TP}}{\text{TP} + \text{FP}}, \\ \text{recall} &= \frac{\text{TP}}{\text{TP} + \text{FN}}, \\ \text{F1} &= 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \end{aligned} \quad (8)$$

4. Experiments and Results

This study will compare the results of the network architectures DenseNet201, ResNet152V2, MobileNetV3, and VGG19 in the cases before and after data normalization with the ADAM optimization function. Then, the study will specifically compare the performance of the above algorithms with the ADAM and ADAS optimization functions on the same normalized dataset.

4.1. Collecting and Normalizing Data

4.1.1. Collecting Data. In this study, the dataset is a set of MRI brain tumors of 123 patients with brain tumors at Bach Mai Hospital, Hanoi, Vietnam, of all ages. Initially, the MRI image was in DICOM format; to remove the information in the patient's DICOM image and convert the image format for machine learning, the DICOM format was converted to the JPEG image format. The size of the converted images is 256×256 pixels.

The image used during training is a T2 pulse sequence image as in Figure 3. Signal intensity with T2 phase correlates very well with not only homogeneity but also tissue profile. Specifically, with low-intensity signals, the tumor has a fibrous and stiffer character than the normal parenchyma. For example, the tumor is fibroblastic in nature, while the more intense sections show a softer characteristic such as a vascular tumor. Therefore, the image of the T2 pulse sequence is considered a pulse sequence that best assesses whether the patient has a brain tumor or not.

TABLE 3: Confusion matrix.

Actual	Predicted	
	Positive	Negative
Positive	TP	FN
Negative	FP	TN

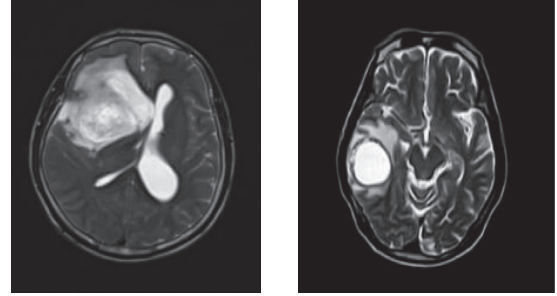


FIGURE 3: Image of the T2 pulse sequence showing the patient's brain tumor.

With the above 123 patients with brain tumor pathology and 100 healthy persons, 1307 images of T2 pulse sequence were selected, of which 647 images showed brain tumors and 660 images did not show brain tumors. The images are all brought to a size of 256×256 pixels to serve the training and testing process of the algorithms.

4.1.2. Normalizing Data

(1) *Minimizing Image Redundancy.* In the raw MRI image data, it is easy to see that there is a rather large black border, but that is the air in the optical field of the machine, so it does not carry information about the skull to be examined. Therefore, it is really necessary to remove the black out-of-the-edge image from the MRI image without affecting the image information content.

The skull on an MRI is usually surrounded by a bright white border, the outer layer of fat around the skull. Meanwhile, the MRI image is a grayscale image (one-dimensional); the range of values of each element in the image matrix representing the brightness of the pixel is in the range $[0, 255]$. In order to maximize the black border on the image, the easy method implemented by this study is to find the first pixel with a nonzero value in the directions from left to right, from right to left, from top to bottom, and from bottom to top as shown in Figure 4. After determining the coordinates of those pixels, remove the outer edges. The normalization of images by cutting out the parts that do not make sense in image classification aims to increase the accuracy of the training process and reduce the training time of the algorithm.

(2) *Normalizing Image Size.* The normalization of the image size helps to improve the accuracy and efficiency of the algorithm. In this study, the image size is 256×256 . This is the right image size for AI algorithms and ensures MRI

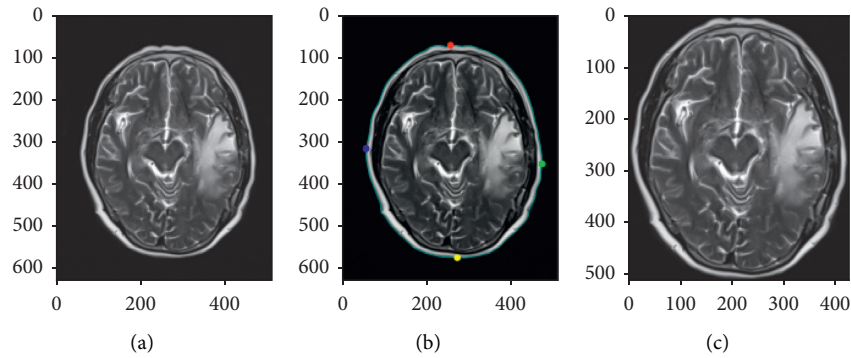


FIGURE 4: Removal of redundant areas of MRI images.

image quality after resizing. Choosing a smaller size will make it difficult for AI algorithms to detect small differences between pixels, affecting the accuracy of the algorithm. If the image size is larger, it will affect the quality of the MRI image after resizing/reducing image quality, negatively affecting the accuracy and performance of the algorithm.

Normalization of data is processed by image data files corresponding to each type of patient's MRI image and by using Python programming. The normalized data removes the nonsignificant parts of the image classification, which increases the accuracy of the model training process and reduces the training time of the algorithm.

4.2. Image Classification Process

Step 1. Preparing the training dataset and feature extraction.

This step is considered an important step in machine learning problems because it is the input for learning to find the model of the problem. We must know how to select the good features, remove the bad features of the data or the noisy components, and estimate how many dimensions of the data are good or in other words how many features to select. If the number of dimensions is too large, making it difficult to calculate, it is necessary to reduce the number of dimensions of the data while maintaining the accuracy of the data (reduce dimension).

In this step, the dataset to test on the model is needed to be prepared. Usually, cross-validation will be used to divide the dataset into two parts, one for training (training dataset) and the other for testing purposes on the model (testing dataset). There are two ways commonly used in cross-validation: splitting and k -folding. For the above algorithms, during the training process, the data is divided according to the ratio 6 : 2 : 2, in which 60% of the data is for training and 20% is for the training validation process (validation). And the remaining 20% is for the process of retesting the model after training.

With the dataset consisting of 1307 images (T2-Images) as mentioned above, the image set has been divided according to the ratio 6 : 2 : 2 to serve the training, validation, and testing processes. Specifically, the number of images used includes 813 images for training, of which 414 images do not show brain tumors and 399 images show

brain tumors; 239 images for validation, including 121 images showing brain tumors and 118 images not showing brain tumors; 255 images for the test process, including 130 images showing brain tumors and 125 images not showing brain tumors.

Step 2. Classifier model.

The purpose of the training model is to find a function $f(x)$ from which to label the data. This step is often called learning or training.

$$f(x) = y, \quad (9)$$

where x is the feature or input of the data and y is the class label or output.

The classification model used here is the above supervised learning algorithms DenseNet201, ResNet152V2, MobileNetV3, and VGG19.

Step 3. Checking data with model to make prediction.

After finding the classification model in Step 2, in this step, new data will be added to test on the classification model.

Step 4. Evaluating the classification model and selecting the best model.

In the final step, the model will be evaluated by assessing the error level of the testing data and the training data through the found model. If the system results are not as expected, the parameters (turning parameters) of the learning algorithms must be changed to find a better model as well as to test and reevaluate the classification model. From there, it is possible to choose the best classification model for the problem. All steps mentioned above can be described as in Figure 5.

4.3. Evaluation of Experiment Results

4.3.1. Evaluating the Effectiveness of Applying Data Normalization

(1) *Results of Training Process.* In order to appraise the effectiveness of data normalization, this work evaluates the convergence (accuracy) of network architectures in

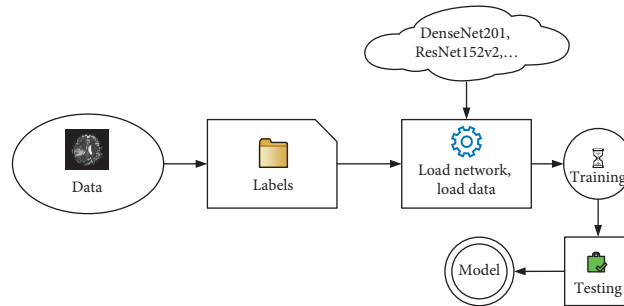


FIGURE 5: The process of creating a model of the algorithm.

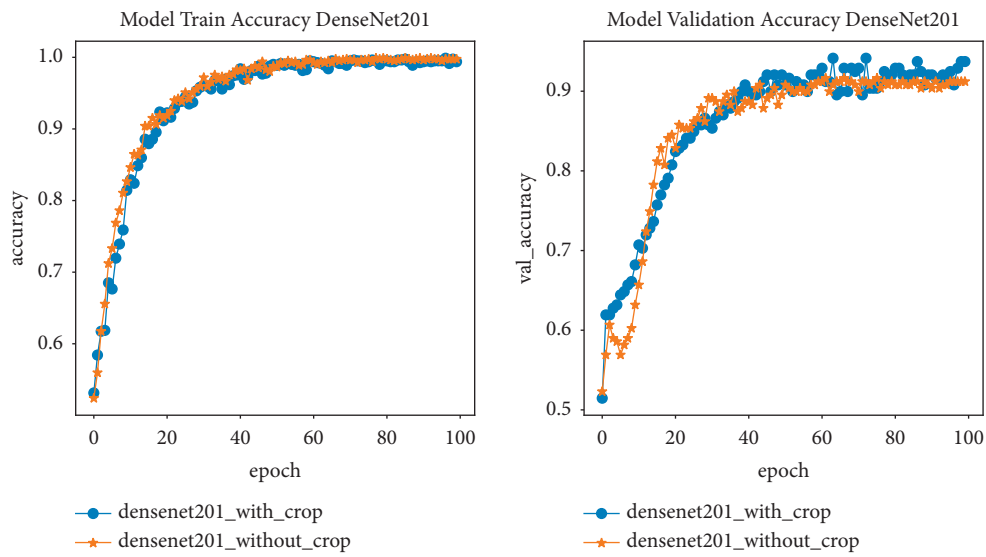


FIGURE 6: Accuracy of DenseNet201 network architecture before and after normalization.

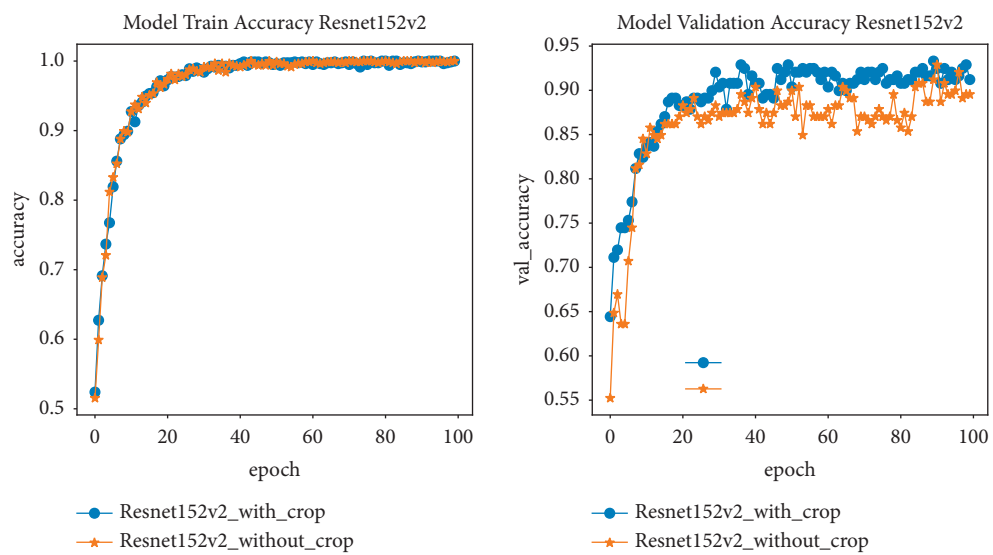


FIGURE 7: Accuracy of ResNet152V2 network architecture before and after normalization.

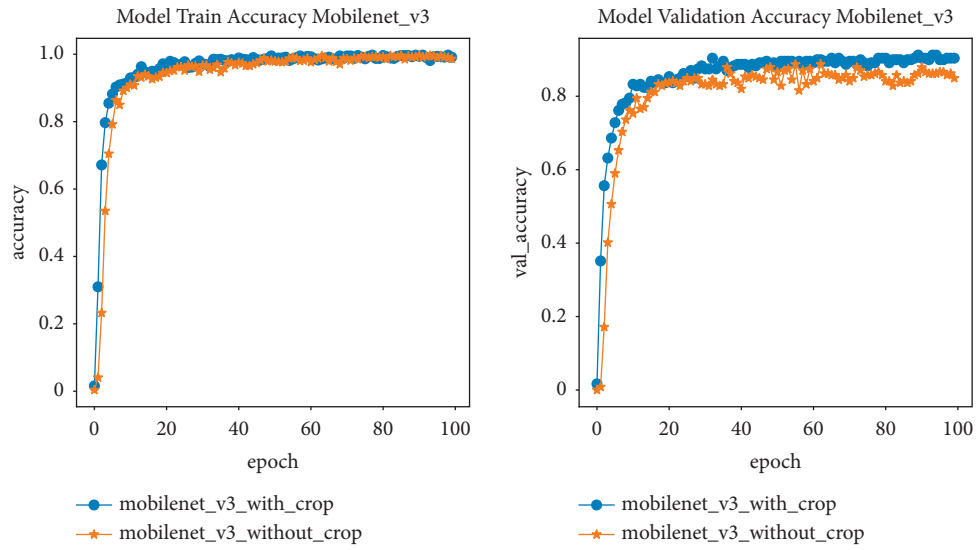


FIGURE 8: Accuracy of MobileNetV3 network architecture before and after normalization.

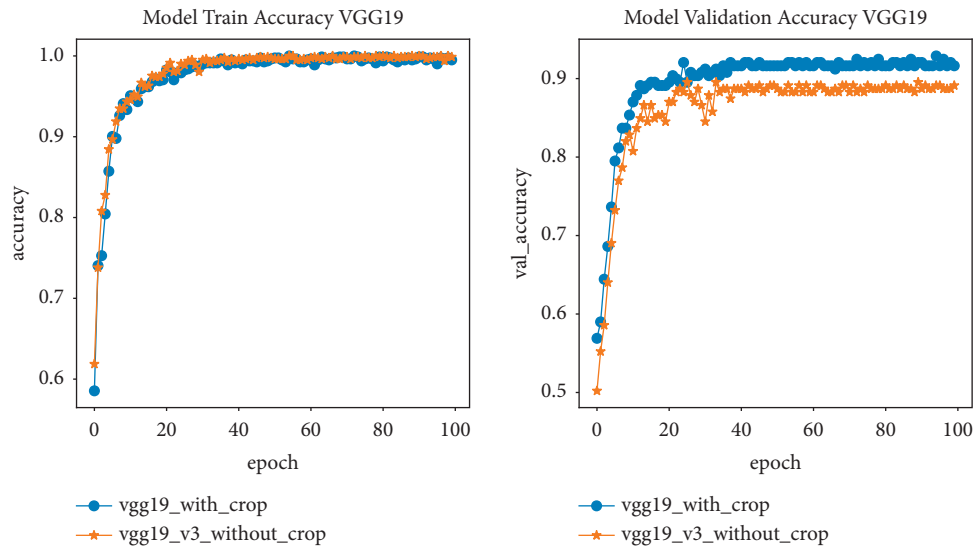


FIGURE 9: Accuracy of VGG19 network architecture before and after normalization.

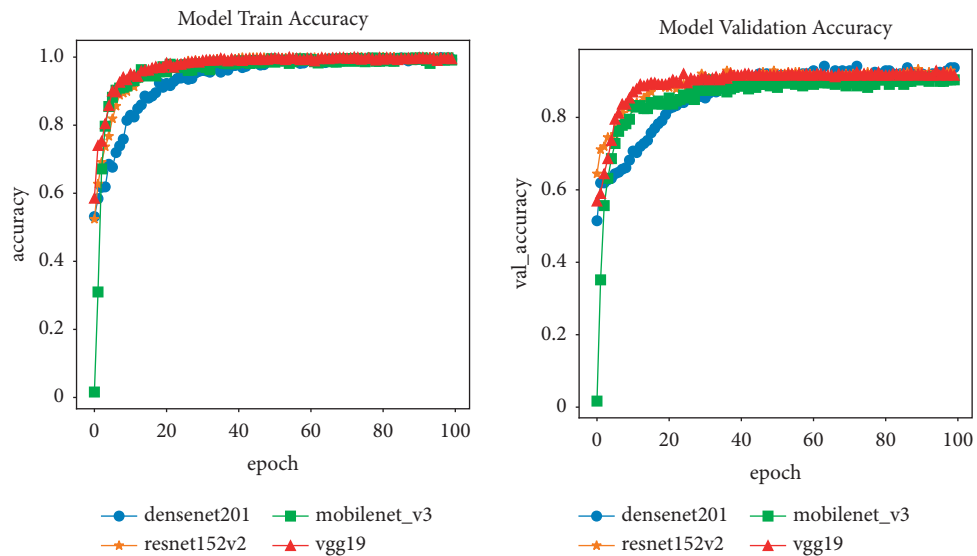


FIGURE 10: Accuracy of network architectures after data normalization.

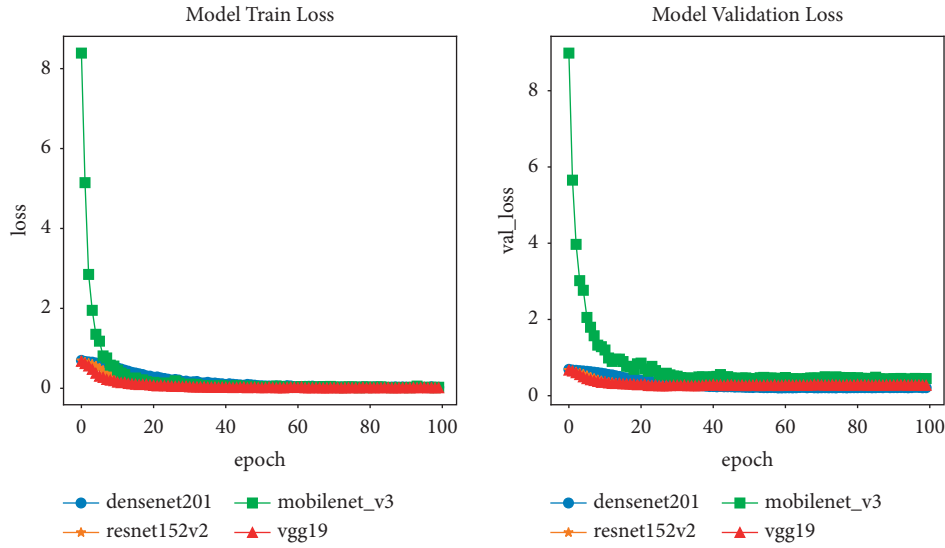


FIGURE 11: Loss of network architectures after data normalization.

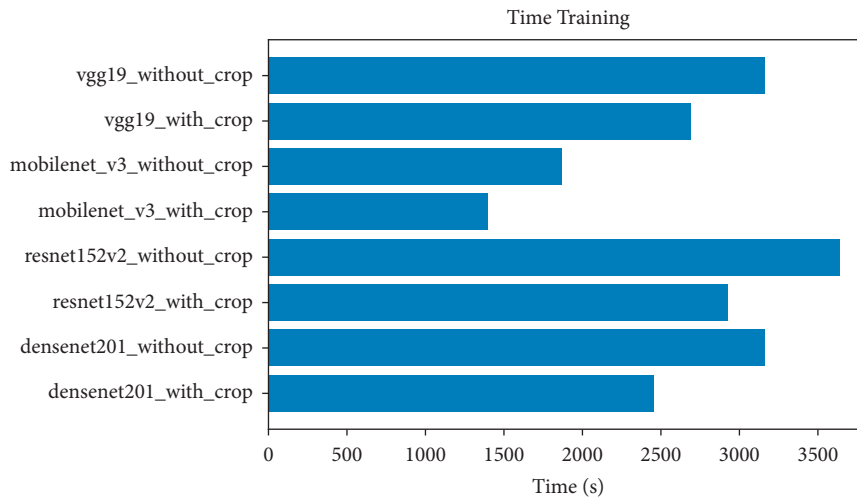


FIGURE 12: Training time over 100 epochs of network architectures before and after normalization.

TABLE 4: Table of comparison results between algorithms before and after data normalization.

CNN architectures	Training accuracy		Validation accuracy		Time (s)	
	Before data normalization	After data normalization	Before data normalization	After data normalization	Before data normalization	After data normalization
DenseNet201	99.87%	99.88%	91.63%	94.14%	3165	455
ResNet152V2	99.95%	99.95%	92.86%	93.31%	3639	927
MobileNetV3	99.66%	99.70%	88.70%	91.21%	1869	1398
VGG19	99.91%	99.91%	89.54%	92.88%	3165	2691

classification through 100 epochs. The network architectures used here are DenseNet201, ResNet152V2, VGG19, and MobileNetV3. The optimal algorithm used in the training processes in this section is the ADAM optimization algorithm.

From Figures 6–9, it is easy to see that the training accuracies before and after normalization of the network architectures are almost the same. The specific results of the DenseNet201, ResNet152V2, MobileNetV3, and VGG19 network architectures are 99.88%, 99.95%, 99.7%, and

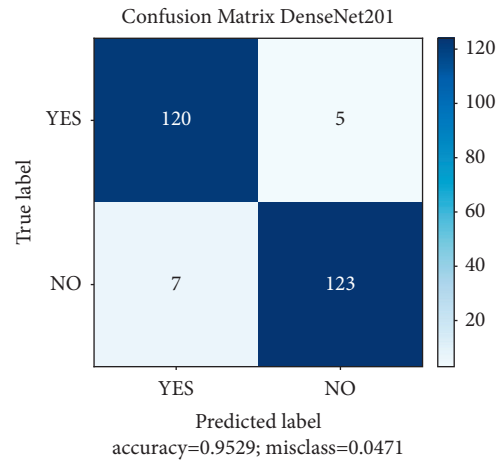


FIGURE 13: DenseNet201 network architecture accuracy based on F1-score.

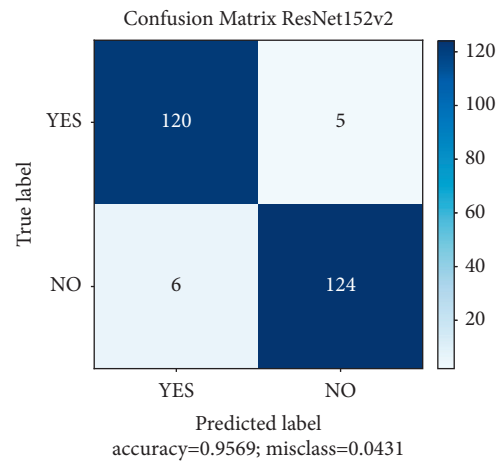


FIGURE 14: ResNet152V2 network architecture accuracy based on F1-score.

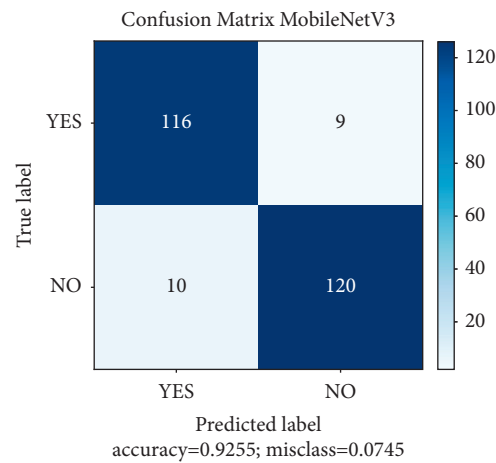


FIGURE 15: MobileNetV3 network architecture accuracy based on F1-score.

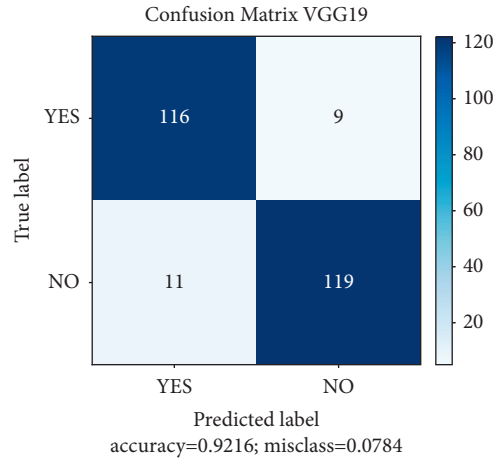


FIGURE 16: VGG19 network architecture accuracy based on F1-score.

TABLE 5: Accuracy of network architectures based on F1-score.

CNN architectures	DenseNet201	ResNet152V2
F1-score	95.29%	95.69%
CNN architectures	MobileNetV3	VGG19
F1-score	92.55%	92.16%

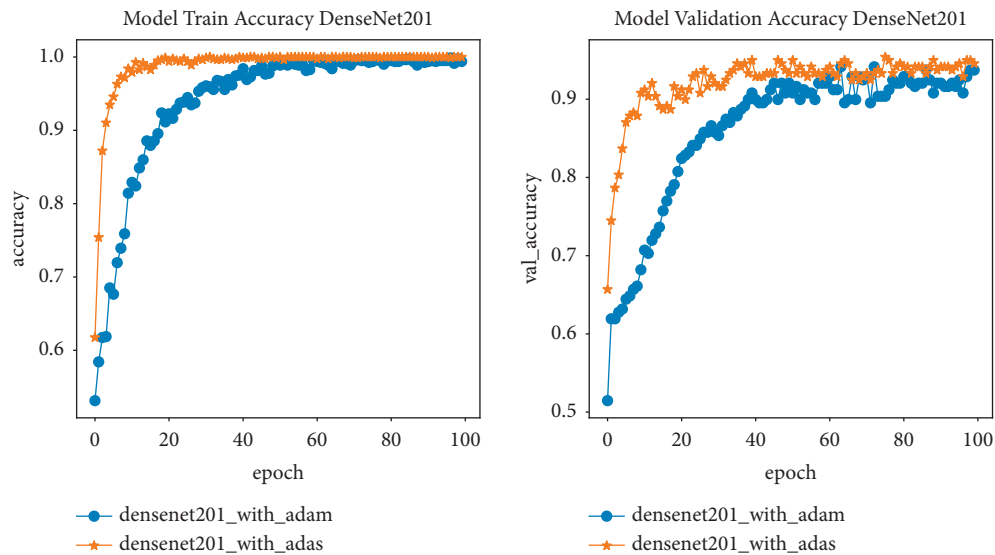


FIGURE 17: Accuracy of DenseNet201 network architectures using ADAM and ADAS optimization algorithms.

99.91%, respectively. However, the results of the validation step of the algorithms showed a marked increase in the accuracy when comparing before and after data normalization. Specifically, the accuracy of the validation process for the DenseNet201 network architecture after normalization is 94.14%, higher than before normalization with an accuracy of 91.63%. The validation result of ResNet152V2 network architecture after normalization has an accuracy of 93.31%, slightly better than before normalization with an accuracy of 92.86%. And this result of ResNet152V2 network architecture after normalization has higher stability than before

normalization as presented in Figure 7; Figure 8 indicates that the validation process of MobileNetV3 network architecture after normalization has higher accuracy than before normalization with accuracy of 91.21% and 88.70%, respectively. The validation results of the VGG19 network architecture are similar to those of the three algorithms above with an accuracy of 92.88% after normalization compared to 89.54% before normalization as shown in Figure 9. And it can be seen that all network architectures have convergence with 90% accuracy after only 40 epochs when they use normalized data.

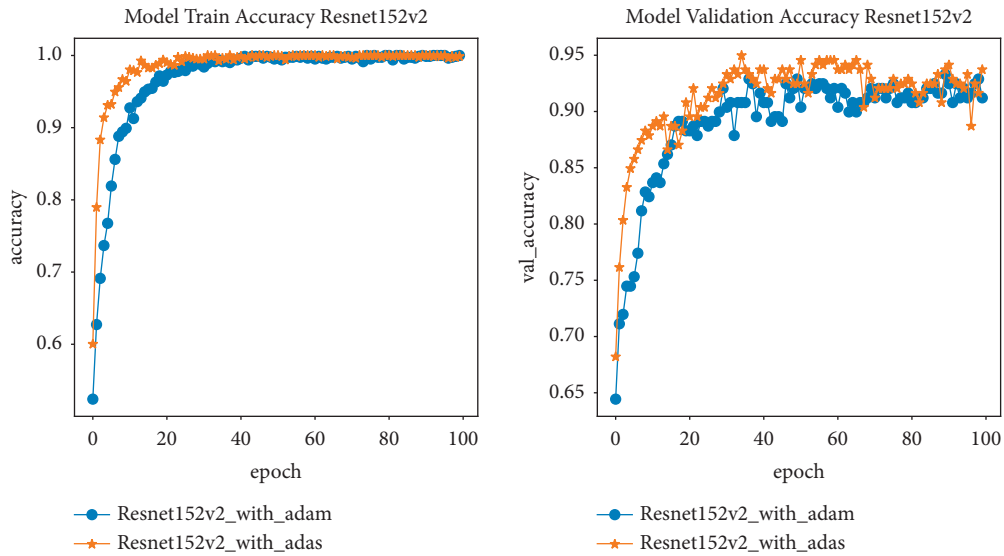


FIGURE 18: Accuracy of ResNet152V2 network architectures using ADAM and ADAS optimization algorithms.

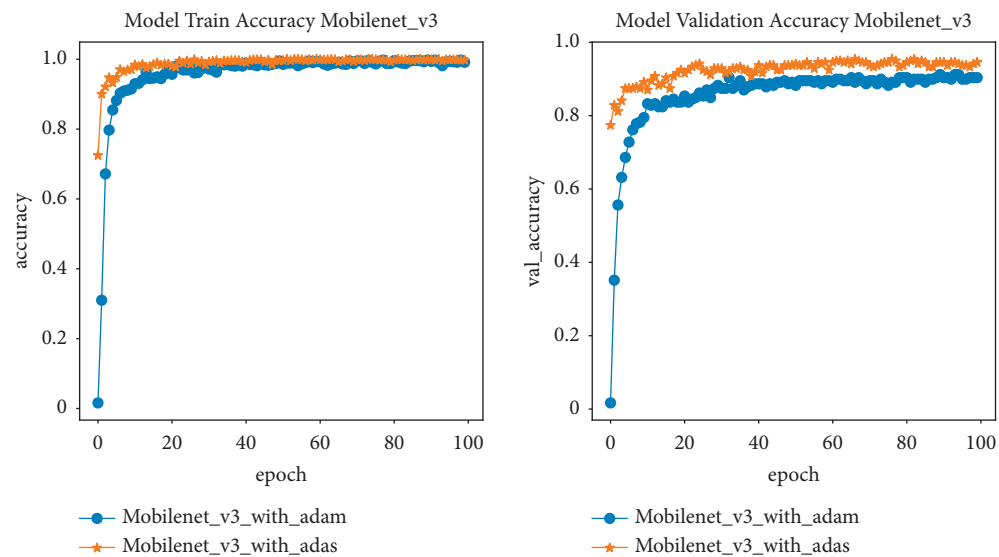


FIGURE 19: Accuracy of MobileNetV3 network architectures using ADAM and ADAS optimization algorithms.

In this paper, in order to be consistent with the collected brain MRI image data, with the ADAM optimal algorithm and the training process with steadily increasing accuracy, where the loss (loss) decreases the most, this study used different learning rates for each network architecture. Specifically, with the network architectures DenseNet201, ResNet152V2, MobileNetV3, and VGG, the initial learning coefficients are $\eta_0 = \{3e - 6; 3e - 6; 2e - 5; 3e - 6\}$, respectively. And the results of using these learning coefficients have shown the stability of the training process to avoid overfitting and are shown in Figures 10 and 11.

In practice, it is not always the case that the longer the model training process, the lower the loss function. When it reaches a certain number of epochs, the loss function value

will reach saturation; it can no longer decrease and may even increase again. That is overfitting phenomenon. To prevent this phenomenon and free up computational resources, the training process should be stopped right at that saturation point. In this study, as shown in Figure 11, it can be seen that the values of the loss function for all architectures reach saturation when the number of epochs is 100.

When comparing the efficiency of processing speed on the same resource, based on Figure 12, it can be seen that all 4 network architectures give a shorter training time with normalized image data than the training time with denormalized image data. Comparison results between algorithms with datasets before and after normalization are shown in Table 4. Clearly, the results showed that the benefits of

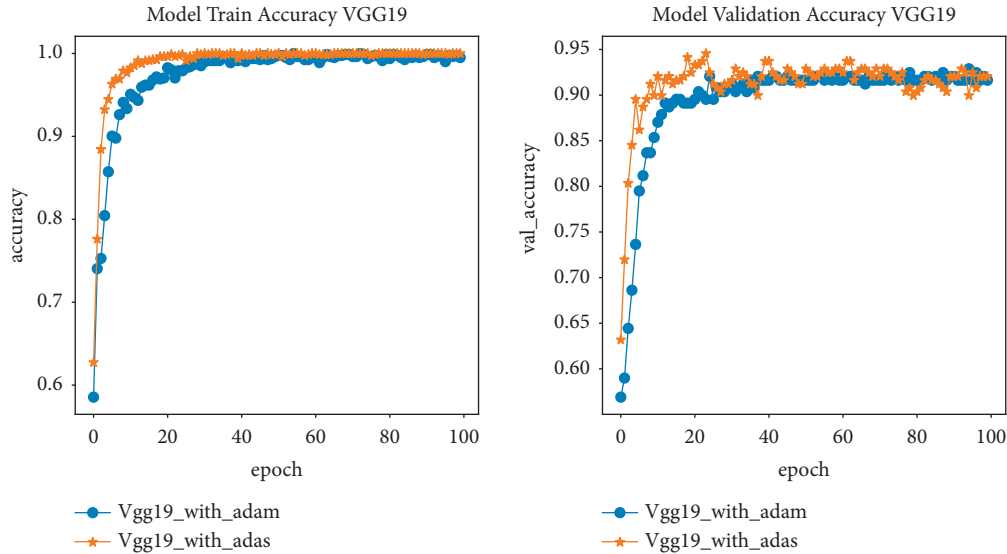


FIGURE 20: Accuracy of VGG19 network architectures using ADAM and ADAS optimization algorithms.

normalizing the image data make the network architectures capable of classifying brain tumors with higher accuracy and shorter training time.

(2) *Evaluating the Accuracy of Network Architectures Based on F1-Score.* After performing the training process, models of the respective network architectures were generated. In this part, they will be tested for testing on the test dataset. This dataset includes 255 images of which 130 images are showing brain tumors and 125 images are not showing brain tumors.

The results illustrated in Figures 13–16 and the summary data in Table 5 show that all algorithms have an accuracy greater than 92% when based on the F1-score, in which ResNet152V2 network architecture has the highest results. This is expected to be implemented in practice.

4.3.2. Comparing the Accuracy of Models Using ADAM and ADAS Optimal Function

(1) *Results of Training Process.* In this section, the accuracy of the classification network architectures will be evaluated and compared using the ADAM and ADAS optimization functions. The network architectures will execute the training, validation, and testing processes on the same normalized database with the same computational resources.

Similar to the ADAM optimization algorithm, in order to fit the brain MRI image data, it is suitable for the ADAS optimization algorithm and the training process has a steady increase in accuracy and the most uniform decrease in loss. Each network architecture uses its own learning coefficient. In this study, the learning coefficients of DenseNet201, ResNet152V2, MobileNetV3, and VGG network architectures are $\eta_0 = \{7e - 3; 5e - 3; 4e - 3; 1e - 2\}$, respectively.

With the above input data, the experimental results of network architectures with ADAM and ADAS optimal

TABLE 6: Comparison results between network architectures using ADAM and ADAS optimization algorithms.

CNN architectures	Training accuracy		Validation accuracy		Times (s)	
	ADAM	ADAS	ADAM	ADAS	ADAM	ADAS
DenseNet201	99.87%	99.88%	94.14%	95.39%	2455 s	2340 s
ResNet152V2	99.95%	99.97%	93.31%	94.47%	2927 s	3161 s
MobileNetV3	99.66%	99.71%	91.21%	95.39%	1398 s	1234 s
VGG19	99.91%	99.9%	92.88%	94.56%	2691 s	2447 s

functions are shown in Figures 17–20, respectively. These results show that the training accuracy of the network architectures using the ADAM and ADAS optimization algorithms are almost the same with the obtained values being greater than 99%. However, for the results of the validation process of the network architectures, the accuracy when implementing the ADAS optimization algorithm has improved significantly in comparison with when using the ADAM optimal algorithm. Specifically, the accuracy of the validation process for the DenseNet201 network architecture using the ADAS optimization algorithm is 95.39% compared to 94.14% when using the ADAM optimization algorithm. And to achieve accuracy, with the ADAM optimal algorithm, the DenseNet201 network needs 40 epochs while with the ADAS optimization algorithm it only needs 10 epochs. The training validation process of ResNet152V2 network architecture using ADAS and ADAM optimization algorithms has the accuracy of 94.47% and 93.31%, respectively. To achieve 90% accuracy, ResNet152V2 network needs 30 epochs when using ADAM optimal algorithm while with ADAS optimal algorithm it only needs 20 epochs. For the MobileNetV3 network architecture, the accuracy of the validation process when using the ADAS optimization function is 95.39% compared to 91.21% when using the ADAM optimization algorithm. The convergence speed for using the ADAS optimization function is also much higher than using the ADAM function, specifically to achieve 90%

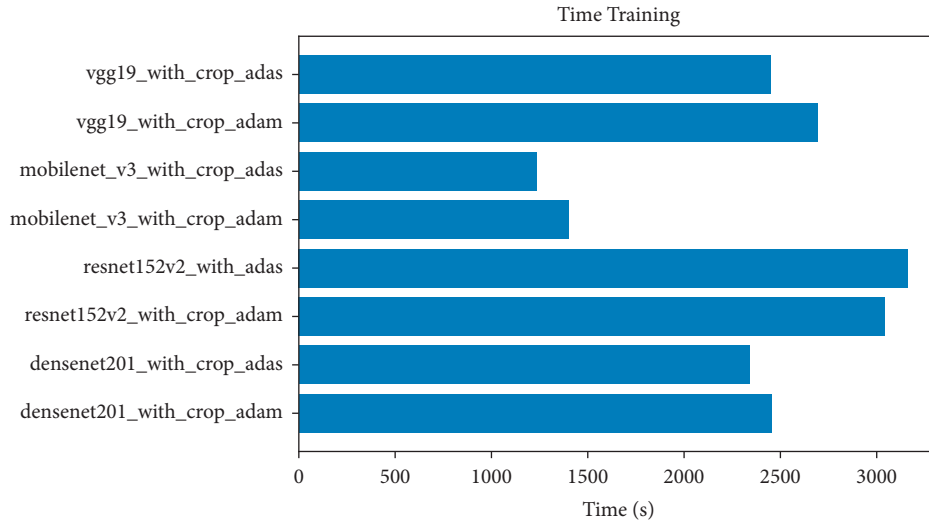


FIGURE 21: Training time over 100 epochs of network architectures with ADAM and ADAS optimization functions.

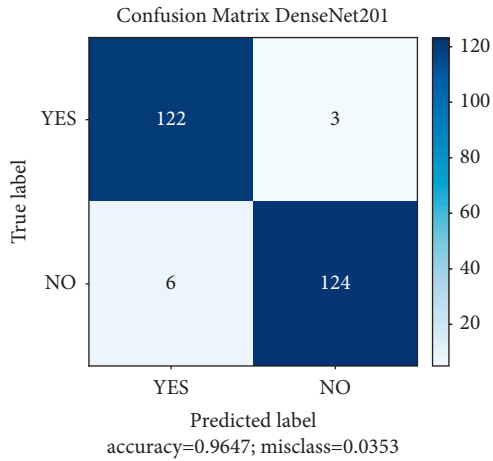


FIGURE 22: Accuracy evaluation matrix of DenseNet201 network architecture using ADAS optimization algorithm via F1-score.

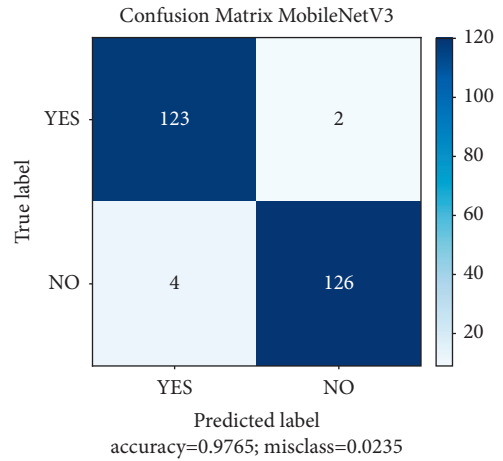


FIGURE 24: Accuracy evaluation matrix of MobileNetV3 network architecture using ADAS optimization algorithm via F1-score.

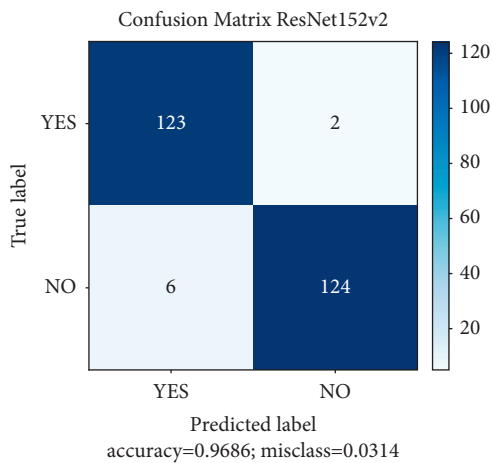


FIGURE 23: Accuracy evaluation matrix of ResNet152V2 network architecture using ADAS optimization algorithm via F1-score.

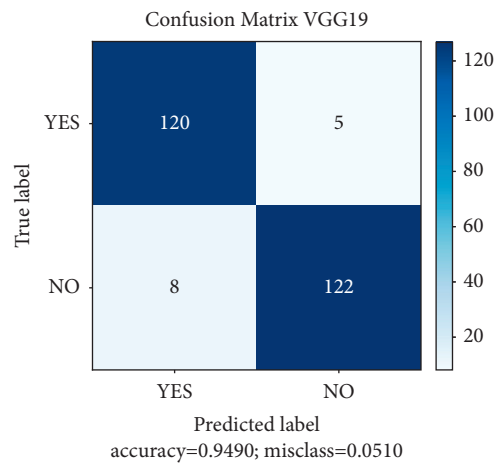


FIGURE 25: Accuracy evaluation matrix of VGG19 network architecture using ADAS optimization algorithm via F1-score.

TABLE 7: Accuracy of network architectures based on F1-score using ADAS optimization function.

CNN architectures	DenseNet201	ResNet152V2
F1-score	96.47%	96.86%
CNN architectures	MobileNetV3	VGG19
F1-score	97.65%	94.90%

TABLE 8: Accuracy and F1-score comparison with other previous studies.

Paper	Algorithms	Accuracy	F1-score
Díaz-Pernas et al. 2021 [18]	Multipathway CNN	99.4%	97.3%
Siddiaue et al. 2021 [16]	Proposed DCNN model	96%	97%
Abd El Kader et al. 2021 [19]	Proposed differential deep-CNN	99.25%	95.23%
Tazin et al. 2021 [20]	CNN architectures	Up to 92%	Up to 92%
This paper	CNN architectures	Up to 99.97%	Up to 97.5%

accuracy, with the ADAM MobileNetV3 function requiring 40 epochs while only 10 epochs are required when using the ADAS function. The VGG19 architecture also has the same results as the above architectures with the accuracy of 94.56% and 92.88%, respectively, with the ADAS and ADAM functions. And also with 90% accuracy, the number of VGG19 network architecture epochs needs to be 25 and 11 when using the ADAM and ADAS functions, respectively.

The performance comparison between ADAS and ADAM algorithms is summarized in Table 6. According to this table as well as the above analysis, it is easy to see that the ADAS optimization algorithm has increased the accuracy of the training process; the convergence in the training process also occurs faster. Figure 21 shows the comparison of training time when using 2 optimization functions with the same normalized dataset. Obviously, the model training time when using the ADAS function in most network architectures is faster. Only for ResNet152V2 architecture, the training time with the use of the ADAS function is slightly longer than with the use of the ADAM function. This can also be one of the problems that need to be studied in the future.

(2) *Evaluation of F1-Score of Network Architectures Using ADAS Optimization Algorithm.* Performing evaluation through F1-score similar to ADAM's algorithm, according to Figures 22–25, the accuracy evaluation through F1-score of network architectures using ADAS optimization function is established as shown in Table 7.

Obviously, when comparing the synthetic results presented in Tables 5 and 7, it is easy to see that the ADAS optimization algorithm has significantly increased the accuracy of the aforementioned models, in which the MobileNetV3 network model gives the highest accuracy of 97.65%. Combined with the results analyzed above, for the problem of brain tumor identification on MRI-T2 images, the ADAS optimization algorithm has significantly improved the accuracy of the training, validation, and testing processes of all the models surveyed in this work as well as

shortening the training time of those models compared to the ADAM algorithm.

4.3.3. *Comparison of Results.* The performance of the proposed system in our study will be compared with the most recently published studies mentioned above. The results of that comparison are shown in Table 8. Based on this table, it is easy to see that the proposed system gave better results in both accuracy and F1-score than other studies with the same subjects. Obviously, although using the same variants of the DCNNs family, the data normalization and the ADAS optimization function helped to significantly improve the performance of the proposed system compared to those other systems.

5. Conclusion

This article has focused on deploying the application of artificial intelligence algorithms in classifying brain tumor patients and normal people using human brain MRI images. The dataset used is MRI images of Vietnamese people, including 123 patients and 100 healthy people. The four algorithms that are experimentally compared in the study are DenseNet201, ResNet152V2, MobileNetV3, and VGG19. The experimental results in the study have shown that the normalization of the initial data processing is very important when it has significantly increased the accuracy in classifying and detecting patients as well as reducing the training time of those models. On the other hand, the paper has also shown the efficiency of the ADAS optimization function compared with the very popular ADAM optimization function. In particular, the ADAS algorithm has advantages in comparison with the ADAM function in improving accuracy as well as reducing model training time. Of the four algorithms mentioned above, the MobileNetV3 algorithm is the most efficient. This can be considered as the foundation for implementing the above system in practice. However, the system also has the disadvantage that the dataset is still small. In the future, besides

collecting more data to increase the accuracy of the system, the research will also develop methods to specifically classify those tumor types according to their tumor characteristics (benign or malignant) or by type of disease.

Data Availability

The data used to support the findings of this study are available upon request from the corresponding author.

Ethical Approval

This study was approved by the Ethics Committee of the Radiology Center, Bach Mai Hospital (Vietnam), and Hanoi University of Science and Technology (Vietnam).

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This research was funded by the Ministry of Education and Training (MOET) Vietnam (Grant no. B2021-BKA-09). The authors also appreciate the support from Hanoi University of Science and Technology (Vietnam) and the Radiology Center of Bach Mai Hospital (Vietnam) in the implementation of this study.

References

- [1] J. C. Buckner, P. D. Brown, B. P. O'Neill, F. B. Meyer, C. J. Wetmore, and J. H. Uhm, "Central nervous system tumors," *Mayo Clinic Proceedings*, vol. 82, no. 10, 2007.
- [2] A. Lashkari, "A neural network based method for brain abnormality detection in MR images using Gabor wavelets," *International journal of computer Applications*, vol. 4, no. 7, pp. 9–15, 2010.
- [3] M. Vargo, "Brain tumor rehabilitation," *American Journal of Physical Medicine & Rehabilitation*, vol. 90, no. 5, pp. S50–S62, 2011.
- [4] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, November 2017.
- [5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, December 2016.
- [6] 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>.
- [7] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, <https://arxiv.org/abs/1409.1556>.
- [8] S. Khawaldeh, U. Pervaiz, A. Rafiq, and R. S. Alkhaldeh, "Noninvasive grading of glioma tumor using magnetic resonance imaging with convolutional neural networks," *Applied Sciences*, vol. 8, no. 1, p. 27, 2018.
- [9] J. S. Paul, A. J. Plassard, B. A. Landman, and F. Daniel, "Deep learning for brain tumor classification," in *Proceedings of the Medical Imaging 2017: Biomedical Applications in Molecular, Structural, and Functional Imaging*, vol. 10137, International Society for Optics and Photonics, Bellingham, WA, USA, March 2017.
- [10] N. Varuna Shree and T. N. R. Kumar, "Identification and classification of brain tumor MRI images with feature extraction using DWT and probabilistic neural network," *Brain informatics*, vol. 5, no. 1, pp. 23–30, 2018.
- [11] D. J. Hemanth, J. Anitha, A. Naaji, O. Geman, D. E. Popescu, and L. Hoang Son, "A modified deep convolutional neural network for abnormal brain image classification," *IEEE Access*, vol. 7, pp. 4275–4283, 2019.
- [12] S. Deepak and P. M. Ameer, "Brain tumor classification using deep CNN features via transfer learning," *Computers in Biology and Medicine*, vol. 111, Article ID 103345, 2019.
- [13] S. Das, O. F. M. R. R. Aranya, and N. N. Labiba, "Brain tumor classification using convolutional neural network," in *Proceedings of the International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, IEEE, Dhaka, Bangladesh, December 2019.
- [14] Z. Ullah, M. U. Farooq, S.-H. Lee, and D. An, "A hybrid image enhancement based brain MRI images classification technique," *Medical Hypotheses*, vol. 143, Article ID 109922, 2020.
- [15] A. Çinar and M. Yildirim, "Detection of tumors on brain MRI images using the hybrid convolutional neural network architecture," *Medical Hypotheses*, vol. 139, Article ID 109684, 2020.
- [16] M. A. B. Siddiaue, S. Sakib, M. M. R. Khan, A. K. Tanzeem, M. Chowdhury, and N. Yasmin, "Deep convolutional neural networks model-based brain tumor detection in brain MRI images," in *Proceedings of the Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, IEEE, Palladam, India, November 2020.
- [17] P. Saxena, A. Maheshwari, and S. Maheshwari, "Predictive modeling of brain tumor: a Deep learning approach," *Innovations in Computational Intelligence and Computer Vision Advances in Intelligent Systems and Computing*, Springer, Berlin, Germany, pp. 275–285, 2021.
- [18] F. J. Díaz-Pernas, M. Martínez-Zarzuela, M. Antón-Rodríguez, and D. González-Ortega, "A deep learning approach for brain tumor classification and segmentation using a multiscale convolutional neural network," *Healthcare*, vol. 9, no. 2, 2021.
- [19] I. Abd El Kader, G. Xu, Z. Shuai, S. Samin, I. Javaid, and I. Salim Ahmad, "Differential deep convolutional neural network model for brain tumor classification," *Brain Sciences*, vol. 11, no. 3, p. 352, 2021.
- [20] T. Tazin, S. Sarker, P. Gupta et al., "A robust and novel approach for brain tumor classification using convolutional neural network," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 2392395, 11 pages, 2021.
- [21] J. Seetha and S. S. Raja, "Brain tumor classification using convolutional neural networks," *Biomedical and Pharmacology Journal*, vol. 11, no. 3, pp. 1457–1461, 2018.
- [22] N. M. Balasooriya and R. D. Nawarathna, "A sophisticated convolutional neural network model for brain tumor classification," in *Proceedings of the IEEE International Conference on Industrial and Information Systems (ICIIS)*, IEEE, Peradeniya, Sri Lanka, February 2017.
- [23] P. Mildnerberger, M. Eichelberg, and E. Martin, "Introduction to the DICOM standard," *European Radiology*, vol. 12, no. 4, pp. 920–927, 2002.
- [24] A. Haase, "Snapshot flash mri. applications to t1, t2, and chemical-shift imaging," *Magnetic Resonance in Medicine*, vol. 13, no. 1, pp. 77–89, 1990.

- [25] X. Zhu and A. B. Goldberg, "Introduction to semi-supervised learning," *Synthesis lectures on artificial intelligence and machine learning*, vol. 3, no. 1, pp. 1–130, 2009.
- [26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.
- [27] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, October 2015.
- [28] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July 2017.
- [29] M. Tan and Q. Le, "Efficientnet: rethinking model scaling for convolutional neural networks," in *Proceedings of the International Conference on Machine Learning*, PMLR, Long Beach, CA, USA, May 2019.
- [30] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár, "Designing network design spaces," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, June 2020.
- [31] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," 2015, <https://arxiv.org/abs/1511.08458>.
- [32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: a large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Miami, FL, USA*, June 2009.
- [33] S. Ruder, "An overview of gradient descent optimization algorithms," 2016, <https://arxiv.org/abs/1609.04747>.
- [34] L. Bottou, "Stochastic gradient descent tricks," *Neural Networks: Tricks of the Trade*, Springer, Berlin, Heidelberg, pp. 421–436, 2012.
- [35] M. S. Hosseini and K. N. Plataniotis, "Adas: adaptive scheduling of stochastic gradients," 2020, <https://arxiv.org/abs/2006.06587>.
- [36] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, p. 7, 2011.
- [37] N. Jegadeesh and S. Titman, "Momentum," *Annual Review of Financial Economics*, vol. 3, no. 1, pp. 493–509, 2011.
- [38] D. P. Kingma and Ba. Jimmy, "Adam: a method for stochastic optimization," 2014, <https://arxiv.org/abs/1412.6980>.
- [39] C. Goutte and E. Gaussier, "A probabilistic interpretation of precision, recall and F-score, with implication for evaluation," in *Proceedings of the European Conference on Information Retrieval*, Santiago de Compostela, Spain, March 2005.