Predicting Brain Tumor Using Transfer Learning

Mustafa Abdul Salam[1,2,*†], Sanaa Taha[3††] , Sameh Alahmady[3††] and Alwan Mohamed[4††]

[1]Artificial Intelligence Department,Faculty of Computers and Artificial Intelligence Benha University, Egypt
[2]Faculty of Computer Studies, Arab Open University, Cairo, Egypt
[3]Faculty of Computers and Information, Egyptian E-Learning UniversityEgypt.
[4]Scientific Computing Department, Faculty of Computers and Artificial Intelligence Cairo University, Egypt

Summary
Brain tumors can also be an abnormal collection or accumulation of cells in the brain that can be life-threatening due to their ability to invade and metastasize to nearby tissues. Accurate diagnosis is critical to the success of treatment planning, and resonant imaging is the primary diagnostic imaging method used to diagnose brain tumors and their extent. Deep learning methods for computer vision applications have shown significant improvements in recent years, primarily due to the undeniable fact that there is a large amount of data on the market to teach models. Therefore, improvements within the model architecture perform better approximations in the monitored configuration. Tumor classification using these deep learning techniques has made great strides by providing reliable, annotated open data sets. Reduce computational effort and learn specific spatial and temporal relationships. This white paper describes transfer models such as the MobileNet model, VGG19 model, InceptionResNetV2 model, Inception model, and DenseNet201 model. The model uses three different optimizers, Adam, SGD, and RMSprop. Finally, the pre-trained MobileNet with RMSprop optimizer is the best model in this paper, with 0.995 accuracies, 0.99 sensitivity, and 1.00 specificity, while at the same time having the lowest computational cost.

Keywords:
Brain Tumor, Transfer Learning, Deep Learning, Computer Vision, MRI.

1. Introduction

According to the global health organization`s statistics, cancer is considered the second leading reason behind human fatalities across the world. Among different types of cancers, the tumor is seen as one of the deadliest, because of its aggressive nature, heterogeneous characteristics, and low relative survival rate. [1] A brain tumor can drastically influence the standard of life, for both patients and their families. The key thing about treating brain cancer and increasing its survivability rate is early diagnosis and properly determining its type. A tumor can have differing kinds (e.g., Meningioma, Pituitary, and Glioma) looking at several factors like the form, texture, and placement of the tumor. Determining the correct tumor type is very important as it can have a significant impact on treatment choices and can predict patient survival. Diagnosis of brain tumors usually includes resonance imaging and biopsy. MRI is recommended because it is non-invasive. However, in some cases, MRI alone is not enough to identify the type of tumor that requires a biopsy. The risks associated with biopsy are high and do not guarantee accurate results. Technicians who perform these steps will have a positive impact on the results and will introduce human error issues. We need a computerized system to help doctors make the right decisions. In recent years, much research has been done on this using various machine learning techniques. Prior to the advent of deep learning, feature selection techniques such as PCA and DWT were used, followed by classifiers such as SVM and ANN. Currently, the first focus is to use neural networks to achieve better results.[2]The prognosis of a brain tumor depends on many factors, including the location of the tumor, the histological subtype of the tumor, and the margin of the tumor. State-of-the-art imaging techniques such as MRI can be used for multiple diagnostic purposes. They can be used to study tumor progression and to identify tumor sites used for surgical prior planning. MR imaging is also used to study anatomy, physiology, and metabolic activity of lesions along with their hemodynamics. Therefore, MR images remain the primary diagnostic modality of brain tumors. [3] Cancer detection, especially early detection, can make a difference in treatment. Early detection is very important because early-stage lesions are likely to heal. Therefore, early intervention can mean the difference between life and death. Deep learning techniques help automate the process of detecting and classifying brain lesions. Also, prioritizing only malignant lesions can reduce the burden on the radiologist to read many images. This ultimately improves overall efficiency and reduces diagnostic errors6. Recent studies have shown that deep learning methods in the field of radiology have already

Manuscript received May 5, 2023
Manuscript revised May 20, 2023
https://doi.org/10.22937/IJCSNS.2023.23.5.10

achieved comparable superhuman performance in some diseases.[4].

1.1. Motivation and contribution

Generally, Brain tumors are an abnormal collection of cells that grow in brain tissue. Benign brain tumors are cured by surgery, but malignant brain tumors are one among the foremost deadly kinds of cancer and may cause direct death. Automatic aim to detect tumors will become an excellent solution to the problem of brain tumors. However, studies face some problems:

Brain tumors may be divided into benign and malignant tumors.

Traditional machine learning requires the existence of a large amount of historical data to guide the model to show acceptable accuracy.

A biopsy may be the only way to make a definitive diagnosis of a brain tumor. A neurosurgeon will perform a biopsy and a pathologist will make a final diagnosis to determine whether the tumor is benign or malignant and grade it accordingly.

The proposed model compared the transfer learning models in terms of prediction accuracy, prediction loss, and prediction time. It is applied to the Br35H:: Brain Tumor Detection 2020 dataset, and the comparison results are reported.

1.2. Paper Organization

The paper is structured as follows: Section 2 devoted for related work. Section 3 presents approaches that describe transfer models; Section 4 reflects on the implementation of models; Section 5 presents on the experiential setup; Section 6 presents the results; while Section 7 points out the core conclusions of the proposed model and highlights the future work.

2. Related Work

Due to the deadly nature of brain tumors, much research has been done to automate their detection and classification. With advances in machine learning, neural networks are gaining attention in developing models for diagnosing brain tumors. Transfer learning techniques can be applied to these models and can be used for other similar diagnoses [5]. This paper attempts to discusses some techniques developed for the classification of brain tumors. Further research and improvement of the technique in this regard is still needed to enable the system developed in to be deployed for physician use.

Muhammad Sjad et. al [6] announced a new multigrade brain tumor classification system based on a convolutional neural network (CNN). Tumor area is segmented using InputCascade CNN, which consists of two separate streams, to extract local and global features. The rich data requirements of the deep learning model are met by applying eight different extended methods with a total of 30 parameters. The pre-trained VGG19CNN architecture is optimized for tumor grade classification. The original and expanded data achieved 87% and 90% accuracy, demonstrating the impact of the data expansion, respectively.

Amin Kabir Anaraki et. al [7] proposed the idea of further developing a CNN architecture for tumor classification using genetic algorithms. This study uses a gadolinium-enhanced T1 image with a size of 128x128 pixels. Simple techniques such as rotation, scaling, and mirroring are applied to increase the size of the dataset. GA is implemented to select parameters such as the number of convolution layers and maximum pooling layers, the number of filters and their size. Bagging of the best model ensemble method developed by GA is used to reduce the variance of the classification error. With this technique, in addition to 5 layers of convolution and maximal pooling, 1 fully connected layer for glioma grading, 6 layers of convolution and maximal pooling, and meningioma, glioma, An architecture consisting of fully connected layers of 384 neurons for the classification of pituitary tumors was obtained. The accuracy achieved was 90.9% and 94.2% for glioma staging and tumor staging, respectively.

Deepak et. Al [8] adopted the concept of transfer learning for feature extraction of the classification system. As a pretreatment, the MRI image was normalized and reduced to 224x224 pixels. The pre-trained GoogLeNet has been modified to learn function from brain MRI. The extracted features are tested on the SVM and ANN classifier models along with the GoogLeNet softmax layer. Deep Transfer Learned (standalone) model, SVM and ANN

classification accuracy is 92.3%, 97.8%, and 98%, respectively.

Vimal Kurup, et. Al [9] used CapsNet to investigate the impact of pretreatment techniques on the classification of brain tumors. Rotation and patch extraction are the pretreatment steps used. The image is scaled to 28x28 pixels and sent to the capsule network. This architecture consists of a hidden layer with a convolutional feature extraction layer and an encapsulation layer, followed by a fully connected classification layer. CapsNet is applied to the original dataset and provides 87 ° accuracy. Applying the same architecture to the preprocessed data gives an accuracy of 92.6, demonstrating that the accuracy increases as the data is preprocessed.

Zar Nawab Khan Swati et. al [10] used a pre-trained deep CNN model, we propose a block-by-block fine-tuning strategy based on transfer learning. This method is evaluated using the 512 x 512 benchmark T1 weighted contrast magnetic resonance imaging (CEMRI) dataset. The image is resized to 224x224 and normalized. The pre-trained VGG19 architecture is divided into 6 blocks based on the pooling layer to reduce tweak time. A 5-directional cross-validation is used to evaluate performance. The accuracy of the proposed method is 94.82%.

Nyoman Abiwinanda et. Al [11] tried to identify the best CNN architecture for brain tumor classification. Five CNN architectures with different numbers of convolutional layers and fully connected layers are being studied. The CNN architecture, which consists of two convolutional layers with 32 filters, activation (ReLu) and Maxpool, followed by a fully-connected layer with 64 neurons, has 84.19% verification accuracy. It has been shown to be optimal. Two variations of this architecture with 64 and 128 filters in the convolution layer will be considered. The verification accuracy of the 32-filter architecture is the highest of the other variants.

Polly et. Al [12] proposed a cod system for the detection and classification of HGG and LGG tumors. Otsu binarization is applied to convert images to binary files. The segmented image then undergoes feature extraction using the discrete wavelet transform. This not only extracts features, but also reduces noise. Due to the large number of features extracted, we use principal component analysis to reduce the features. The SVM then classifies the image as HGG and LGG. Tested with 100 images, the accuracy of this system is 99%.

Heba Mohsen et. Al [13] considered a deep neural network for classifying 66 brain MRI datasets into four classes. The fuzzy means technique is used to divide an image into five sections. The Discrete Wavelet Transform extracts features from segmented tumor regions. Feature reduction is performed by principal component analysis. The classifiers used are DNN with seven hidden layers, ANN with k = 1 and k = 3, Linear Discriminant Analysis (LDA) and SMOSVM. DNN offers the highest accuracy of any technology at 98.4%.

Garima Singh et. Al [14] proposed a brain tumor classification system using a normalized histogram and segmentation using a K-means clustering algorithm. Weigh various noise reduction techniques such as median filters, adaptive filters, Gaussian filters, averaging filters, and unsharp mask filters. The median filter is selected for use because it provides the highest peak signal-to-noise ratio. After normalizing the histogram, the images are classified into tumor and non-tumor images using a naive Bayes classifier and SVM. SVMs have proven to be more efficient at 91.49% than at 87.23% for Naive Bayes. Images in which tumors were detected were segmented using the K-Means algorithm.

Parnian Afshar et.al [15] proposed a CapsNet architecture for brain tumor classification. The MRI image fed to the network is down sampled from 512x512 to 128x128. The second layer is the convolution layer. Two layers are followed by a layer of capsules, the last layer containing capsules of each tumor type. Tumor boundaries linked to the resulting vector pass through two fully connected layers with different numbers of neurons. Finally, the SoftMax layer returns the probability of each tumor class present. The proposed architecture provides 90.89% accuracy.

Table 1 The related work summary

Author Name	Model	Accuracy
Muhammad Sjad et. Al [6]	Pre – trained VGG – 19 CNN	90%
Amin Kabir Anaraki et. Al [7]	CNN	94.2%
Deepak et. Al [8]	KNN	97.8%
Vimal Kurup, et. Al [9]	CapsNet	92.6%
Zar Nawab Khan Swati et. Al [10]	Pre – trained deep CNN	94.82%
Nyoman Abiwinanda et. Al [11]	CNN	84.19%

F. P. Polly et. Al [12]	SVM	99%
Heba Mohsen et. Al [13]	Deep Neural Network	98.4%
Garima Singh et. Al [14]	SVM	91.49%
Parnian Afshar et.al [15]	CapsNet	90.89%

3. Approach

This section addresses the applied tools and methodologies to detect the tumors state of the subject:

3.1. Image Processing

As shown in fig[1], the dataset raw images.

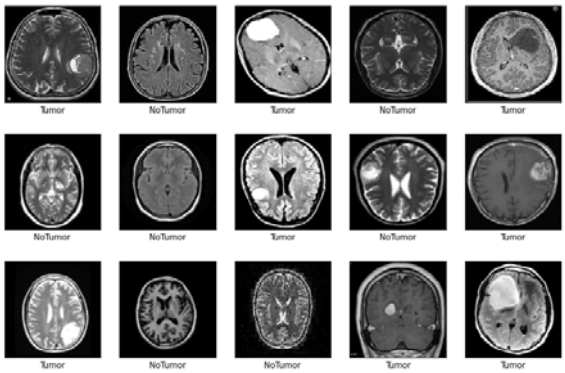

Fig 1 The dataset raw images

As shown in fig [2], the dataset images after image processing applied with histogram equalization. This technique usually enhances the overall contrast of many images, especially when the image is represented by a narrow range of intensity values. This adjustment allows you to use the entire range of intensities evenly and better distribute the intensities on the histogram. As a result, areas with low local contrast can have high contrast. Histogram equalization accomplishes this by effectively distributing the populous intensity values used to reduce the contrast of the image [16].

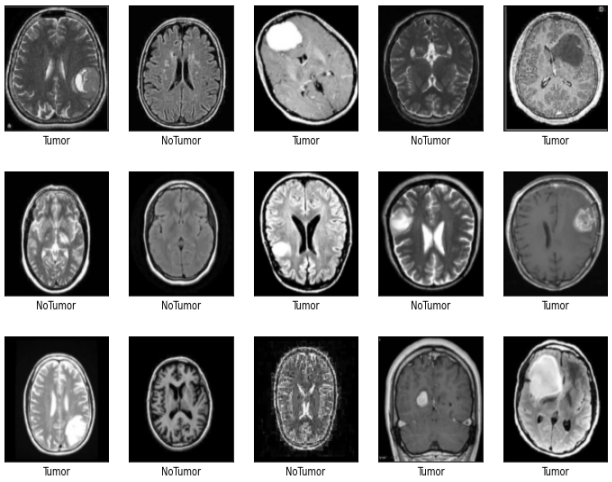

Fig 2 Images after image processing

3.2. MobileNet Model

Introducing a category of efficient models called MobileNets [17] for mobile and embedded image processing applications. MobileNet is predicated on an optimized architecture that uses depth-separable convolution to create lightweight deep neural networks. Here are two simples global hyperparameters that effectively balance latency and accuracy. These hyperparameters allow the modeler to settle on a model of the correct size for the appliance, supporting the constraints of the matter. It presents extensive experiments on resource accuracy trade-offs and demonstrates powerful performance compared to other popular ImageNet classification models. the subsequent shows the effectiveness of MobileNet in an exceeding style of applications and use cases, including object detection, detailed classification, facial attributes, and huge positions.

3.3. VGG19 Model

This model investigates the effect of convolutional network depth on accuracy in large image recognition environments [18]. The biggest contribution is to thoroughly assess the increase in network depth using a very small (3x3) convolution filter architecture. This indicates that increasing the depth to the 1619 weight layer can significantly improve the configuration of the prior art. These results inspired my submission to the ImageNet Challenge 2014. In this challenge, the team took first and second place on the localization track, and the classification track. It also shows that our representation is well generalized to other datasets and provides up-to-date results. To facilitate further research on the use of deep visual

representation in computer vision, we have published two best-performing ConvNet models.

3.4. InceptionResNetV2 Model

Very deep convolutional networks have been central to major advances in image recognition performance in recent years [19]. An example is the Inception architecture. It has been shown to achieve very good performance with relatively low complexity. Recently, with the introduction of residual connections in combination with more traditional architecture, the 2015 ILSVRC Challenge achieved state-of-the-art performance. Its performance was similar to the latest generation Inception v3 network. This raises the question of whether it would be beneficial to combine the Inception architecture with rest connections. Here we provide clear empirical evidence that training with the remaining connections will significantly speed up training on the starting network. There is also evidence that the remaining Inception networks slightly outperform the similarly priced Inception networks with no remaining connections. It also introduces some new architectures optimized for both REST and non-REST starting networks. These variations significantly improve the performance of single-frame detection on ILSVRC2012 classification tasks. In addition, proper activation scaling shows how to stabilize training in a very wide residual initiation network. Using the three residuals and the Inception v4 ensemble, you can achieve a Top 5 error of 3.08 percent in the ImageNet Classification Challenge (CLS) test set.

3.5. Inception Model

Convolutional networks [20] form the core of modern computer vision solutions for a variety of tasks. Since 2014, very deep convolutional networks have become mainstream and various benchmarks have increased significantly. Increasing model size and complexity will quickly improve the standard for most tasks (as long as sufficient labeled data is provided for the training), but computational efficiency and a few parameters are mobile vision. This is still an important factor for various use cases such as big data and large data. Data scenario. Here we consider how to scale the network to use the additional computations as efficiently as possible through properly factorized convolution and aggressive regularization. We compare our method with the ILSVRC2012 Classification Challenge Verification Set and show significant improvements over the prior art. Millions of parameters for frame evaluation using less than 25 networks, with 5 billion multiplications per 21.2% Top1 and 5.6% Top5 error inferences. Report within the validation set (3.6 test set% error) and 17.3% Top 1 error in the validation set using an ensemble of 4 models and a multi-crop evaluation to 3.5% Top 5 error.

3.6. DenseNet201 Model

Recent studies have shown that convolutional networks will be much deeper [21], more accurate, and more efficient in training if they contain short connections between the layers near the doorway and also the layers near the exit. this text takes this observation and introduces the Dense Convolutional Network (DenseNet), which connects all layers to all or any other layers during a feedforward manner. Whereas traditional L-layer convolutional networks have L connections, one between each layer and also the layers that follow it, our network has L (L + 1) / 2 direct connections. Each layer receives a feature map of all previous layers as input, and its own feature map is employed as input for all subsequent layers. DenseNet has some compelling advantages. It mitigates the vanishing gradient problem, enhances feature propagation, facilitates feature reuse, and significantly reduces the number of parameters. Evaluate the proposed architecture for four highly competitive object detection benchmarking tasks (CIFAR10, CIFAR100, SVHN, and ImageNet). DenseNets provide significant improvements over traditional techniques in most cases but require less computing power to attain high performance.

4. PROPOSED MODEL

This section addresses the proposed models to detect the tumors state of the subject.
As shown in Figure [3], the proposed Transfer model building steps are:

- Data Loading: Loading images from directories as class for each directory.
- Apply histogram equalization: Appling image processing using sci-lit images API.
- Split Data: Splitting data to train, test and validate sets.
- Load keras Application: using tf.keras.applications to load required application.
- Load Transfer Model: Downloading the base model from keras API.
- Train and evaluate the model: using of sci-kit learn metrics API to evaluate the results of training.

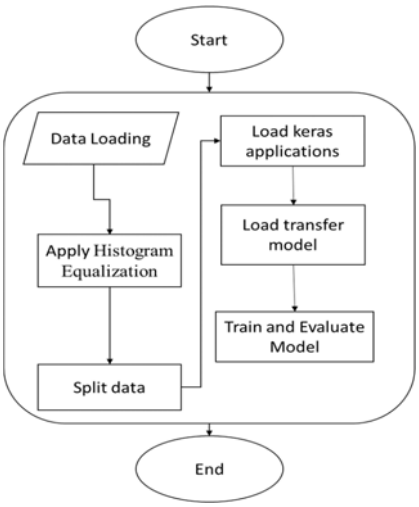

Fig 3 Transfer learning approach

5. Experimental Setup

5.1. Br35H Dataset [22]

Brain tumors are considered one of the most aggressive illnesses for children and adults. Brain tumors account for 85 to 90 percent of all primary CNS diseases of the central nervous system. tumor. About 11,700 people are diagnosed with brain tumors each year. The 5-year survival rate for people with cancerous brain tumors or CNS tumors is about 34 percent for men and 36 percent for girls. Brain tumors are classified into benign tumors, malignant tumors, and pituitary tumors. Proper treatment, planning, and accurate diagnosis are required to boost lifetime in patients. the most effective technique for detecting brain tumors is resonance imaging (MRI). Scanning creates an outsized amount of image data. These images are examined by a radiologist. Due to the complexity and characteristics of brain tumors, manual examinations are often error-prone. Applying automated classification techniques using machine learning (ML) and computer science (AI) is more consistent and accurate than manual classification. Therefore, doctors around the world can propose systems that perform detection and classification using deep learning algorithms using convolutional neural networks (CNN), artificial neural networks (ANN), and transfer learning (TL). It's impossible. Convenient. Dentures. The dataset contains three folders: yes, no, and pred contain 3060 brain MRI images.

5.2. Evaluation criteria

There are only 4 cases any subject could end up with:

- True positive (TP): Prediction is +ve and X has tumor.
- True negative (TN): Prediction is -ve and X has not tumor.
- False positive (FP): Prediction is +ve and X has not tumor.
- False negative (FN): Prediction is -ve and X has tumor.

Accuracy: It's the ratio of the correctly labeled subjects to the entire pool of subjects.

Accuracy answers the subsequent question: what number subjects did correctly label out of all the subjects?

(TP+TN)/(TP+FP+FN+TN) (1)

Precision: It is that the ratio of the correctly +ve labeled by model to all or any +ve labeled.

Precision answers the following: what percentage of these who labeled as yes are literally tumor?

TP/(TP+FP) (2)

Recall "Sensitivity": It is that the ratio of the correctly +ve labeled by our program to all or any who are diabetic really.

Recall answers the subsequent question: Of all the subjects that had tumor, what number of these we correctly predict?

TP/(TP+FN) (3)

F1 Score: It is the harmonic mean(average) of the precision and recall.

F1 Score is best if there's some kind of balance between precision (p) & recall (r) within the system. Oppositely F1 Score isn't so high if one measure is improved at the expense of the opposite.

2*(Recall * Precision) / (Recall + Precision) (4)

Specificity: It is that the correctly -ve labeled by the program to all or any who are healthy really.

Specifity answers the subsequent question: Of all the people that are healthy, what number of these did we correctly predict?

TN/(TN+FP) (5).

6. Results

6.1. **MobileNet Model**

6.1.1. **Adam Optimizer**

As shown in table [2], it presented the metrices evaluated for trained MobileNet optimized by

Adam Optimizer. Then Fig [4], [5] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [6].

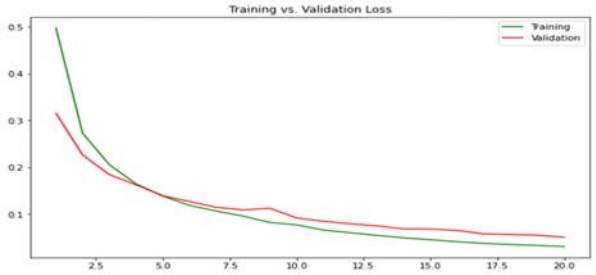

Fig 4 MobileNet training vs validation loss with Adam optimizer

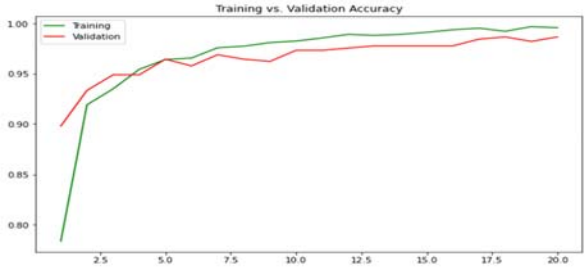

Fig 5 MobileNet training vs validation accuracy with Adam optimizer

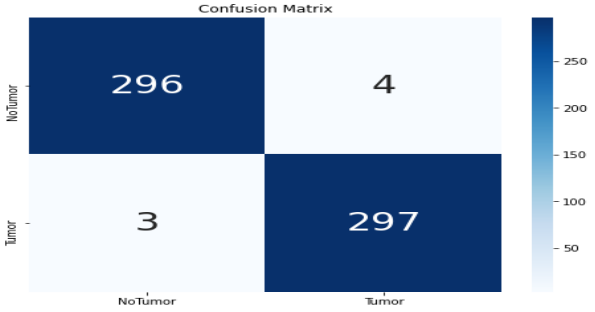

Fig 6 MobileNet confusion matrix with Adam optimizer

Table 2 MobileNet metrices with Adam optimizer

Metric	Performance
Accuracy	98.83
Balanced Accuracy	98.83
Precision	98.67
Recall	99.00
Specificity	98.67
F1-Score	98.84

6.1.2. RMSprop Optimizer

As shown in table [3], it presented the metrices evaluated for trained MobileNet optimized by RMSprop Optimizer. Then Fig [7], [8] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [9].

Fig 7 MobileNet training vs validation loss with RMSprop optimizer

Fig 8 MobileNet training vs validation accuracy with RMSprop optimizer

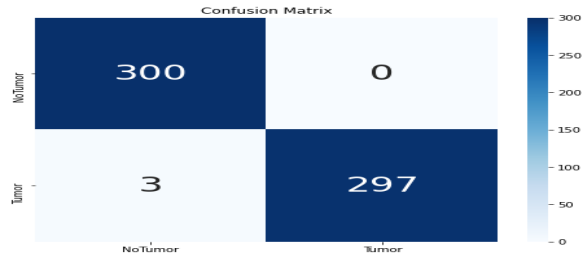

Fig 9 MobileNet confusion matrix with RMSprop optimizer

Table 3 MobileNet metrics with RMSprop optimizer

Metric	Performance
Accuracy	99.5
Balanced Accuracy	99.5
Precision	100.0
Recall	99.0
Specificity	100.0
F1-Score	99.5

6.1.3. SGD Optimizer

As shown in table [4], it presented the metrices evaluated for trained MobileNet optimized by SGD Optimizer. Then Fig [10], [11] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [12].

Fig 10 MobileNet training vs validation loss with SGD optimizer

Fig 11MobileNet training vs validation accuracy with SGD optimizer

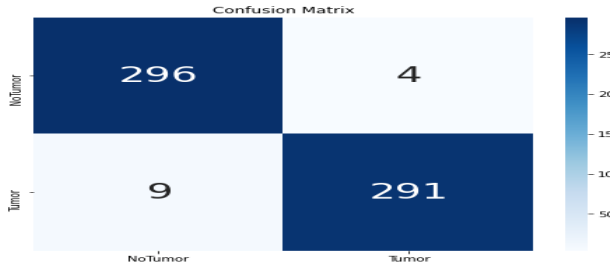

Fig 12 MobileNet confusion matrix with SGD optimizer

Table 4 MobileNet metrices with SGD optimizer

Metric	Performance
Accuracy	97.83
Balanced Accuracy	97.83
Precision	98.64
Recall	97.0
Specificity	98.67
F1-Score	97.82

6.2. VGG19 Model

6.2.1. Adam Optimizer

As shown in table [5], it presented the metrices evaluated for trained VGG19 optimized by Adam Optimizer. Then Fig [13], [14] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [15].

Fig 14 VGG19 training vs validation loss with Adam optimizer

Fig 15 VGG19 training vs validation accuracy with Adam optimizer

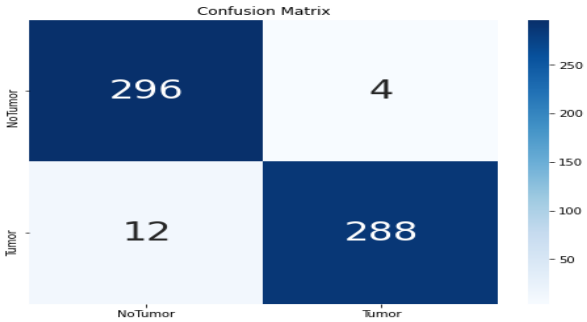

Fig 13 VGG19 confusion matrix with Adam optimizer

Table 5 VGG19 metrices with Adam optimizer

Metric	Performance
Accuracy	97.33
Balanced Accuracy	97.33
Precision	98.63
Recall	96.0
Specificity	98.67
F1-Score	97.30

6.2.2. RMSprop Optimizer

As shown in table [6], it presented the metrices evaluated for trained VGG19 optimized by RMSprop Optimizer. Then Fig [16], [17] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [18].

Fig 16 VGG19 training vs validation loss with RMSprop optimizer

Fig 17 VGG19 training vs validation accuracy with RMSprop optimizer

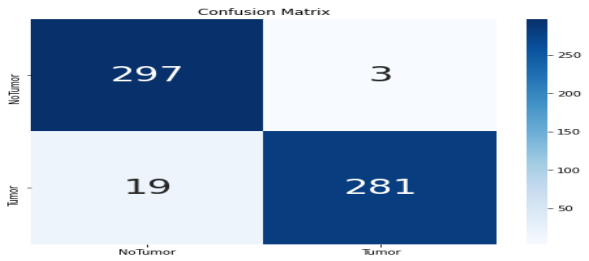

Fig 18 VGG19 confusion matrix with RMSprop optimizer

Table 6 VGG19 metrices with RMSprop optimizer

Metric	Performance
Accuracy	96.33
Balanced Accuracy	96.33
Precision	98.94
Recall	93.67
Specificity	99.00
F1-Score	96.23

6.2.3. SGD Optimizer

As shown in table [7], it presented the metrices evaluated for trained VGG19 optimized by SGD Optimizer. Then Fig [19], [20] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [21].

Fig 19 VGG19 training vs validation loss with SGD optimizer

Fig 20 VGG19 training vs validation accuracy with SGD optimizer

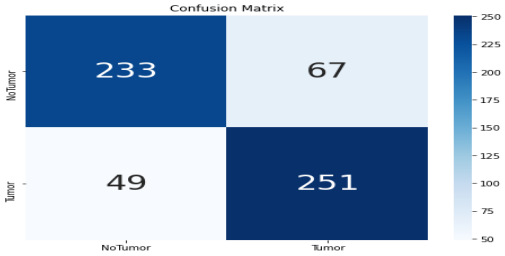

Fig 21 VGG19 confusion matrix with SGD optimizer

Table 7 VGG19 metrices with SGD optimizer

Metric	Performance
Accuracy	80.67
Balanced Accuracy	80.67
Precision	78.93
Recall	83.67
Specificity	77.67
F1-Score	81.23

6.3. InceptionResNetV2 Model

6.3.1. Adam Optimizer

As shown in table [8], it presented the metrices evaluated for trained InceptionResNetV2 optimized by Adam Optimizer. Then Fig [22], [23] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [24].

Fig 22 InceptionResNetV2 training vs validation loss with Adam optimizer

Fig 23 InceptionResNetV2 training vs validation accuracy with Adam optimizer

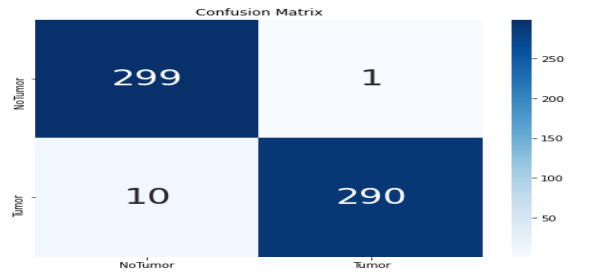

Fig 24 InceptionResNetV2 confusion matrix with Adam optimizer

Table 8 VGG19 metrices with Adam optimizer

Metric	Performance
Accuracy	98.17
Balanced Accuracy	98.17
Precision	99.66
Recall	96.67
Specificity	99.67
F1-Score	98.14

6.3.2. RMSprop Optimizer

As shown in table [9], it presented the metrices evaluated for trained InceptionResNetV2 optimized by RMSprop Optimizer. Then Fig [25], [26] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [27].

Fig 25 InceptionResNetV2 training vs validation loss with RMSprop optimizer

Fig 26 InceptionResNetV2 training vs validation accuracy with RMSprop optimizer

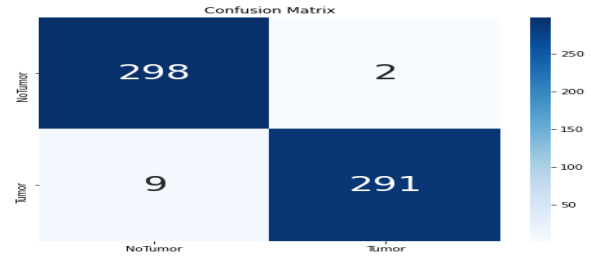

Fig 27 InceptionResNetV2 confusion matrix with RMSprop optimizer

Table 9 InceptionResNetV2 metrices with RMSprop optimizer

Metric	Performance
Accuracy	98.17
Balanced Accuracy	98.17
Precision	99.32
Recall	97.00
Specificity	99.33
F1-Score	98.15

6.3.3. SGD Optimizer

As shown in table [10], it presented the metrices evaluated for trained InceptionResNetV2 optimized by SDG Optimizer. Then Fig [28], [29] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [30].

Fig 28 InceptionResNetV2 training vs validation loss with SGD optimizer

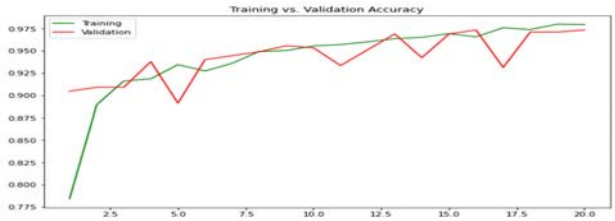

Fig 29 InceptionResNetV2 training vs validation accuracy with SGD optimizer

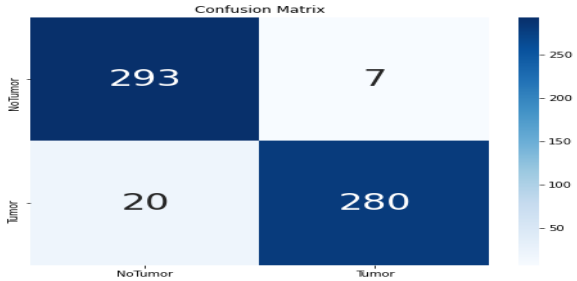

Fig 30 InceptionResNetV2 confusion matrix with SGD optimizer

Table 10 InceptionResNetV2 metrices with SGD optimizer

Metric	Performance
Accuracy	95.5
Balanced Accuracy	95.5
Precision	97.56
Recall	93.33
Specificity	97.67
F1-Score	95.4

6.4. Inception Model

6.4.1. Adam Optimizer

As shown in table [11], it presented the metrices evaluated for trained Inception optimized by Adam Optimizer. Then Fig [31], [32] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [33].

Fig 31 Inception training vs validation loss with Adam optimizer

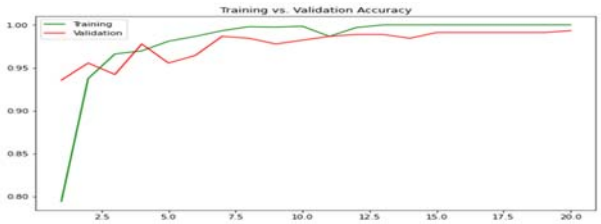

Fig 32 Inception training vs validation accuracy with Adam optimizer

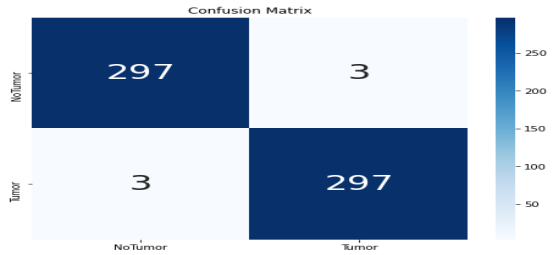

Fig 33 Inception confusion matrix with Adam optimizer

Table 11 Inception metrices with Adam optimizer

Metric	Performance
Accuracy	99.0
Balanced Accuracy	99.0
Precision	99.0
Recall	99.0
Specificity	99.0
F1-Score	99.5

6.4.2. RMSprop Optimizer

As shown in table [12, it presented the metrices evaluated for trained Inception optimized by RMSprop Optimizer. Then Fig [34], [35] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [36].

Fig 34 Inception training vs validation loss with RMSprop optimizer

Fig 35Inception training vs validation accuracy with RMSprop optimizer

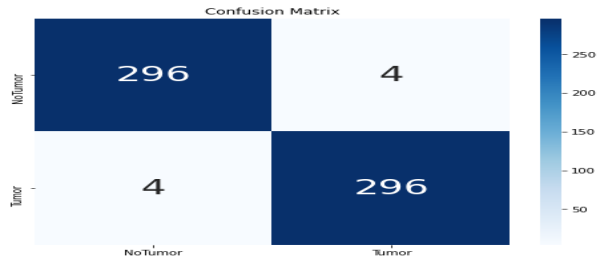

Fig 36 Inception confusion matrix with RMSprop optimizer

Table 12 Inception metrices with RMSprop optimizer

Metric	Performance
Accuracy	98.67
Balanced Accuracy	98.67
Precision	98.67
Recall	98.67
Specificity	98.67
F1-Score	98.67

6.4.3. SGD Optimizer

As shown in table [13], it presented the metrices evaluated for trained Inception optimized by SGD Optimizer. Then Fig [37], [38] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [39].

Fig 37 Inception training vs validation loss with SGD optimizer

Fig 38 Inception training vs validation accuracy with SGD optimizer

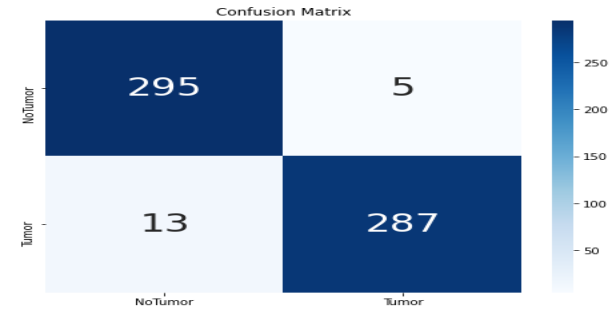

Fig 39 Inception confusion matrix with SGD optimizer

Table 13 Inception metrices with SGD optimizer

Metric	Performance
Accuracy	97.00
Balanced Accuracy	97.00
Precision	98.29
Recall	95.67
Specificity	98.67
F1-Score	96.96

6.5. DenseNet201 Model

6.5.1. Adam Optimizer

As shown in table [14], it presented the metrices evaluated for trained DenseNet201 optimized by Adam Optimizer. Then Fig [40], [41] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [42].

Fig 40 DenseNet201 training vs validation loss with Adam optimizer

Fig 41 DenseNet201 training vs validation accuracy with Adam optimizer

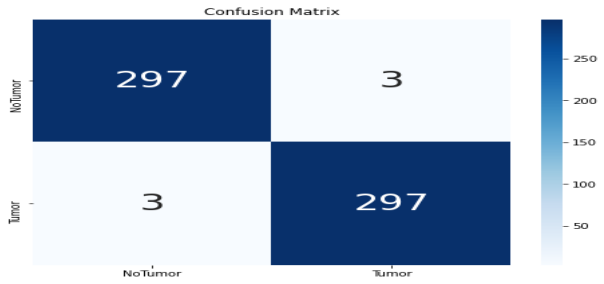

Fig 42 DenseNet201 confusion matrix with Adam optimizer

Table 14 DenseNet201 metrices with Adam optimizer

Metric	Performance
Accuracy	99.0
Balanced Accuracy	99.0
Precision	99.0
Recall	99.0
Specificity	99.0
F1-Score	99.0

6.5.2. RMSprop Optimizer

As shown in table [15], it presented the metrices evaluated for trained DenseNet201 optimized by Adam Optimizer. Then Fig [43], [44] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [45].

Fig 43 DenseNet201 training vs validation loss with Rmsprop optimizer

Fig 44 DenseNet201 training *vs validation accuracy with RMSprop optimizer*

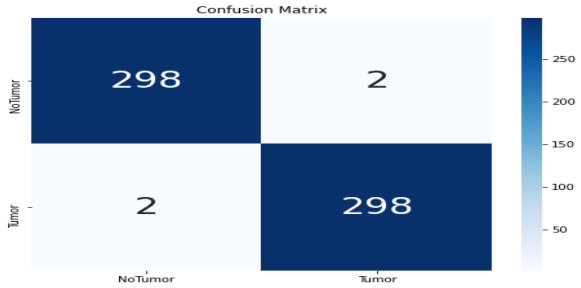

Fig 45 DenseNet201 confusion matrix with RMSprop optimizer

Table 15 DenseNet201 metrices with RMSprop optimizer

Metric	Performance
Accuracy	99.33
Balanced Accuracy	99.33
Precision	99.3
Recall	99.33
Specificity	99.33
F1-Score	99.33

6.5.3. SGD Optimizer

As shown in table [16], it presented the metrices evaluated for trained DenseNet201 optimized by Adam Optimizer. Then Fig [46], [47] shown the training vs validation loss and accuracy respectively. However, confusion matrix is depicting in fig [48].

Fig 46 DenseNet201 training vs validation loss with SGD optimizer

Fig 47 DenseNet201 training vs validation accuracy with SGD optimizer

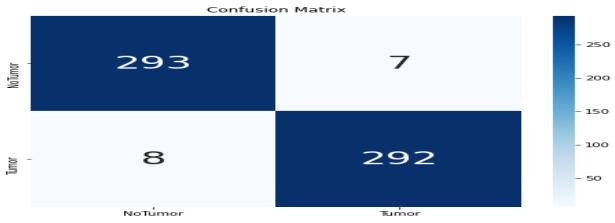

Fig 48 DenseNet201 confusion matrix with SGD optimizer

Table 16 DenseNet201 metrices with SGD optimizer

Metric	Performance
Accuracy	97.5
Balanced Accuracy	97.5
Precision	97.66
Recall	97.33
Specificity	97.67
F1-Score	97.5

6.6. Summary

6.6.1. Accuracy

As shown in table [17], it presented the accuracy comparison between models optimized with 3 different optimizers.

Table 17 The accuracy comparison between models optimized with 3 different optimizers

Model/optimizer	Adam	RMSprop	SGD
MobileNet	98.83	**99.5**	97.83
VGG19	97.33	96.33	80.67
InceptionResNetV2	98.17	98.17	95.50
Inception	99.00	98.67	97.00
DenseNet201	99.00	**99.33**	97.50

6.6.2. Sensitivity

As shown in table [18], it presented the Sensitivity comparison between models optimized with 3 different optimizers.

Table 18 The Sensitivity comparison between models optimized with 3 different optimizers.

Model/optimizer	Adam	RMSprop	SGD
MobileNet	99.00	**99.0**	97.00
VGG19	96.00	93.67	83.67
InceptionResNetV2	96.67	97.00	93.33
Inception	99.00	98.67	95.67
DenseNet201	99.00	**99.33**	97.33

6.6.3. Specificity

As shown in table [19], it presented the Specificity comparison between models optimized with 3 different optimizers.

Table 19 The Specificity comparison between models optimized with 3 different optimizers

Model/optimizer	Adam	RMSprop	SGD
MobileNet	98.67	**100.0**	98.67
VGG19	98.67	99.00	77.67

InceptionResNetV2	99.67	99.33	97.67
Inception	99.00	98.67	98.33
DenseNet201	99.00	99.33	97.67

7. Conclusion and future work

Imaging classification of brain tumors can be an important part of medical imaging. Help doctors make accurate diagnoses and treatment plans. This paper proposes a neoplasmic MR image classification method using transfer learning. Accurate and accurate tumor MR imaging plays an important role in clinical diagnosis and patient treatment. High-end machine learning techniques for classification focus only on low-level or high-level features, use some hand-crafted features to fill this gap and seek superior feature extraction and classification methods. increase. Recent developments in deep learning have made great strides, and deep convolutional neural networks (CNNs) have successfully solved the task of image classification. Deep learning is attractive for feature representations that map low-level and high-level information perfectly and incorporate feature extraction and classification phases into self-learning but usually require a large training dataset. In some medical imaging scenarios, the small data set makes it difficult to use deep learning and training CNN from scratch with a small data set. To solve this problem, we propose a block-by-block fine-tuning strategy supported by transfer learning as the MobileNet model, VGG19 model, InceptionResNetV2 model, Inception model, and DenseNet201 model. The proposed model does not use hand-crafted features, requires minimal pre-processing, and has the highest effective accuracy of 99.5%, 99% sensitivity, and 100% for RMSprop-optimized MobileNet models. It is more common to achieve specificity.

References

[1] Fritz, A. et al. International Classifcation of Diseases for Oncology Vol. 3 (World Health Organization, Geneva, 2001).

[2] Goodenberger, M. L. et al. Genetics of adult glioma. Cancer Genet. 205, 613–621 (2012).

[3] Claus, E. B. et al. Survival and low-grade glioma: Te emergence of genetic information. Neurosurg. Focus 38, E6 (2015).

[4] Menze, B. H. et al. Te multimodal brain tumor image segmentation benchmark (brats). IEEE Trans. Med. 34, 1993–2024 (2014).

[5] Mzoughi, H. et al. Deep multi-scale 3d convolutional neural network (cnn) for mri gliomas brain tumor classifcation. J. Digit. Imaging 33, 903–915 (2020).

[6] Muhammad Sjjad, Salman Khan, Khan Muhammad, Wanqing Wu, Amin Ullah, Sung Wook Baik, “Multi-grade brain tumor classification using deep CNN with extensive data augmentation”, Elsevier, Journal of Computational Science 30, pp 174-182, 2019.

[7] Amin Kabir Anaraki, Moosa Ayati, Foad Kazemi, “Magnetic resonance imaging-based brain tumor grades classification and grading via convolutional neural networks and genetic algorithms”, Elsevier, Biocybergenetics and Biomedical Engineering 39, pp 63-74, 2019.

[8] Deepak, P.M. Ameer, “Brain tumor classification using deep CNN features via transfer learning”, Elsevier, Computers in Biology and Medicine 111, pp 1-7, 2019.

[9] R. Vimal Kurup, V. Sowmya, K. P. Soman, “Effect of Data Pre-processing on Brain Tumor Classification Using Capsulenet”, Springer, ICICCT System Reliability, Quality Control, Safety, Maintenance and Management, pp 110-119, 2019.

[10] Zar Nawab Khan Swati, Qinghua Zhao, Muhammad Kabir, Farman Ali, Zakir Ali, Saeed Ahmed, Jianfeng Lu, “Brain tumor classification for MR images using transfer learning and finetuning”, Elsevier, Computerized Medical Imaging and Graphics 75, pp 34-46, 2019.

[11] Nyoman Abiwinanda, Muhammad Hanif, S. Tafwida Hesaputra, Astri Handayani, and Tati Rajab Mengko, “Brain Tumor Classification Using Convolutional Neural Network”, Springer, World Congress on Medical Physics and Biomedical Engineering, pp 183-189, 2018.

[12] F. P. Polly, S. K. Shil, M. A. Hossain, A. Ayman, and Y. M. Jang, “Detection and Classification of HGG and LGG Brain Tumor Using Machine Learning”, IEEE, International Conference on Information Networking (ICOIN), 2018.

[13] Heba Mohsen, El-Sayed A. El-Dahshan, El-Sayed M. ElHorbaty, Abdel-Badeeh M. Salem, “Classification using deep learning neural networks for brain tumors”, Elsevier, Future Computing and Informatics Journal 3, pp 68-71, 2018.

[14] Garima Singh, Dr M.A.Ansari, “Efficient Detection of Brain Tumor from MRIs Using K-Means Segmentation and Normalized Histogram”, IEEE, 1st India International Conference on Information Processing (IICIP), 2016.

[15] Parnian Afshar, Konstantinos N. Plantaniotis, Arash Mohammadi, “Capsule Networks for brain tumor classification based on MRI images coarse tumor boundaries”, IEEE, International Conference on Acoustics, Speech and Signal Processing, 2019.

[16] Hum, Yan Chai; Lai, Khin Wee; Mohamad Salim, Maheza Irna (October 11, 2014). "Multiobjectives bihistogram equalization for image contrast enhancement". Complexity. 20 (2): 22–36. Bibcode:2014Cmplx..20b..22H. doi:10.1002/cplx.21499.

[17] Howard, Andrew G. and Zhu, Menglong and Chen, Bo and Kalenichenko, Dmitry and Wang, Weijun and Weyand, Tobias and Andreetto, Marco and Adam, Hartwig (2017), MobileNets: Efficient Convolutional Neural Networks for

Mobile Vision Applications, arXiv.org perpetual, non-exclusive license, 10.48550/ARXIV.1704.04861.
[18] Simonyan, Karen and Zisserman, Andrew (2014), Very Deep Convolutional Networks for Large-Scale Image Recognition, arXiv.org perpetual, non-exclusive license, 10.48550/ARXIV.1409.1556.
[19] Szegedy, Christian and Ioffe, Sergey and Vanhoucke, Vincent and Alemi, Alex (2016), Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning, arXiv.org perpetual, non-exclusive license, 10.48550/ARXIV.1602.07261.
[20] Szegedy, Christian and Vanhoucke, Vincent and Ioffe, Sergey and Shlens, Jonathon and Wojna, Zbigniew (2015), Rethinking the Inception Architecture for Computer Vision, arXiv.org perpetual, non-exclusive license, 10.48550/ARXIV.1512.00567.
[21] Huang, Gao and Liu, Zhuang and van der Maaten, Laurens and Weinberger, Kilian Q. (2016), Densely Connected Convolutional Networks, arXiv.org perpetual, non-exclusive license, 10.48550/ARXIV.1608.06993.
[22] https://www.kaggle.com/datasets/ahmedhamada0/brain-tumor-detection.

MUSTAFA ABDUL SALAM Received Ph.D. degree in Hybrid machine learning models with swarm intelligence optimization from the faculty of Computers and Information, Cairo University, Egypt, in 2015. He is currently an Associate Professor with the AI Department, Faculty of Computers and Information, Benha University, Egypt. He has worked on a number of research topics. He has contributed more than 30 technical articles in the areas of ANN, SVM, optimization, time series prediction, ELM, and hybrid CI models in international journals, and international conferences. His majors are hybrid machine learning models, big data analytics, data streams mining, and federated learning

Sanaa Taha received her B.Sc. (2001) and M.Sc. (2005) degrees from the Department of Information Technology, Faculty of Computers and Information, Cairo University, Egypt, and a Ph.D. degree (2013) in Electrical and Computer Engineering from the University of Waterloo, Canada. She is currently an assistant professor in the Department of Information Technology, Faculty of Computer and Information, Cairo University, Cairo, Egypt. Her research interests include wireless network security, mobile networks security, mobility management, and applied cryptography.

Alwan A. Mohamed was born in Cairo, Egypt, in August 1994. She received the B.Sc. degree from the Higher Technological Institute, Department of Computer Science, Egypt, A researcher at master degree at scientific computing department, Benha University, Egypt. She specializing in hybrid machine learning, bio inspired optimization algorithms, and federated learning technology. She is currently a teaching assistant with the Computer Science Department, Modern Academy for computer and management sciences, Egypt. Her majors are machine learning, big data, stream data mining, and deep learning.