Automated Brain Disease Classification using Transfer Learning based Deep Learning Models

Farhana Alam[1], Farhana Chowdhury Tisha[2], Sara Anisa Rahman[3], Samia Sultana[4]
Md. Ahied Mahi Chowdhury[5], Ahmed Wasif Reza[6]*, Mohammad Shamsul Arefin[7]
Department of Computer Science and Engineering, East West University, Dhaka-1212, Bangladesh[1, 2, 3, 4, 5, 6]
Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh[7]
Department of Computer Science and Engineering, Chittagong University of Engineering and Technology[7]
Chattogram, Bangladesh[7]

***Abstract*—Brain MRI (Magnetic Resonance Imaging) classification is one of the most significant areas of medical imaging. Among different types of procedures, MRI is the most trusted one to detect brain diseases. Manual and semi-automated segmentations need highly experienced radiologists and much time to detect the problem. Recently, deep learning methods have taken attention due to their automation and self-learning techniques. To get a faster result, we have used different algorithms of Convolutional Neural Network (CNN) with the help of transfer learning for classification to detect diseases. This procedure is fully automated, needs less involvement of highly experienced radiologists, and does not take much time to provide the result. We have implemented six deep learning algorithms, which are InceptionV3, ResNet152V2, MobileNetV2, Resnet50, EfficientNetB0, and DenseNet201 on two brain tumor datasets (both individually and manually combined) and one Alzheimer's dataset. Our first brain tumor dataset (total of 7,023 images-training 5,712, testing 1,311) has 99-100 percent training accuracy and 98-99 percent testing accuracy. Our second tumor dataset (total of 3,264 images-training 2,870, testing 394) has 100 percent training accuracy and 69-81 percent testing accuracy. The combined dataset (total of 10,000 images-training 8,000, testing 2,000) has 99-100 percent training accuracy and 98-99 percent testing accuracy. Alzheimer's dataset (total of 6,400 images-training 5,121, testing 1,279, 4 classes of images) has 99-100 percent training accuracy and 71-78 percent testing accuracy. CNN models are renowned for showing the best accuracy in a limited dataset, which we have observed in our models.**

***Keywords*—Brain MRI; tumor; deep learning; classification; transfer learning**

I. Introduction

With the advancement of modern science and technology, brain diseases are still among the deadliest diseases. Magnetic Resonance Imaging (MRI) is a well-known term in the medical sector to diagnose cerebral complications. It is used to detect brain cells that differ from normal cells. There are some other methods such as X-radiation (X-rays), Computed Tomography (CT), Positron Emission Tomography (PET), Single-Photon-Emission Computed Tomography (SPECT), Magnetic Resonance Spectroscopy (MRS), etc. are also used for diagnosis of diseases. But among all of them, MRI is the most popular one to detect problematic cells accurately. MRI is a non-invasive and flexible clinical method that investigates the conditions of the brain and any other body parts in species [1]. It uses magnetic fields and radio waves to generate images. For brain MRI, the images are taken from different planes to detect the actual area of the problematic cells of both pre-and post-contrast. Its' scanned images provide high contrast and high spatial resolution images, which helps to understand the different characteristics of the soft tissues of a cell. Usually, brain abnormalities are easily found by MRI scans. After analyzing those images, medical experts can easily identify brain disorders such as Alzheimer's disease, schizophrenia, multiple sclerosis, brain tumors, cancer, and degenerative diseases [1]. Although, many neurological diseases need frequent analysis of the brain, in those cases MRI scan is essential.

In the past, segmentation done by humans was a time-consuming procedure and could not provide significant results [2]. On the other hand, automatic segmentation methods result in efficient and precise segmentation. Lately, deep learning methods have been given increasing attention due to their automation and self-learning techniques. Convolutional Neural Network (CNN) is one of the most popular deep learning architectures and it has shown outstanding impact on various industries, such as medical, electronics, robotics, etc. The main advantage of CNN is that it can learn abstract features of the image without having preceding acknowledgment compared to classical methods. This method is developing daily and has achieved numerous appreciations in brain segmentation and classification. Precise segmentation of a 2D and 3D image has always been a challenging task, and various approaches have been proposed for better accuracy in the past. But state-of-the-art deep learning architectures for image segmentation have managed to compute complex 3D models. For these reasons, automatic detection and classification are highly demanding attributes in the decision-making of medical science. Again, CNN models show high accuracy even in limited datasets which are also one of the reasons for choosing CNN models. As we have used transfer learning, the process has become faster. Most of the traditional supervised learning algorithms are not supportive of multi-class classifications as well as very few experiments have been done on recently developed deep learning algorithms for brain MRI classification. Therefore, the question remains what is an efficient way to classify brain diseases from MR images? Also, a comparison of different deep learning algorithms on different types of datasets is

*Corresponding Author.

missing, which raises the question of how well a model works on different types of images.

The main objective of this study is to find an efficient way to classify diseases from brain MRI using deep learning models and show a comparative study of them for multi-class brain MRI classification problems. Six CNN models which are commonly used in classification of brain MRI- InceptionV3 [3], ResNet152V2, MobileNetV2 [4], Resnet50 [5], EfficientNetB0 [6] and DenseNet201 [7][8][9][10][11][12][13][14][15][16][17][18][19]. We have implemented these models on three different datasets- one is an Alzheimer's dataset, and the others are brain tumor datasets, all of which are open-access datasets. This study contributes to the health sector, where it is crucial to act in a short time in case of any emergency. Our study can reduce the time to classify a disease from an MR image, which can also lower the occurrence of human error. Also, it can reduce the cost for the patients. As CNN models are getting developed day by day, we could improve our health sector services by finding the most efficient one.

The rest of the paper is organized as follows: In Section II, we have reviewed the related paper materials and their research analogy. A brief overview of publicly available brain MRI datasets, followed by a brain MRI analysis and overview of CNN architectures are discussed in Section III. We have analyzed the performance of our proposed architecture on three publicly available datasets and compared their performance with other methods in Section IV. In Section V, we conclude the paper.

II. Related Works

Deep learning models are very recent but many research works have been done for the classification of brain tissues. A method for binary classification of brain tumors is proposed, where they took only the region of interest from MRI images by using Open source Computer Vision (CV) Canny Edge Detection technique and trained a CNN model of eight convolutional layers [20]. Multiclass classification of brain tumors is proposed by selecting features using Densenet201 Pre-Trained Deep Learning Model, Entropy–Kurtosis-based High Feature Values (EKbHFV), and a modified genetic algorithm (MGA), where Cubic SVM classifier is used to classify the selected features after fusing using a non-redundancy-based fusion approach [21]. Again, a CNN model of 18 layers is used for cropped lesions, uncropped lesions, and segmented lesion images for multiclass classification of brain tumors [22].

Some works have either pre-trained data or implemented a single model. MobileNetV2 is used to classify brain tumors with the accuracy of 94% [15], applied ResNet152V2 for classifying four types of brain tumors by using various pre-processing steps to achieve an accuracy of 98.9% [11], and used 29 different pre-trained models to classify Alzheimer's disease, achieved the highest accuracy of 92.98% by EfficientNetB0 [19].

In some literature, they have used multiple planes and multiple layers to detect the problem. A multi-pathway CNN architecture is proposed where the input images are processed in three spatial scales: sagittal, coronal, and axial views [23] and implemented CNN model with small kernels and neuron weight to classify between tumor and non-tumor which brought 97.5% accuracy with very low complexity [24]. Some works have been done by summing up a few models. Using a method where pre-trained models are used for feature concatenation, it is found that features from the pre-trained model of InceptionV3 and DensNet201 can classify three-class brain tumor datasets better than existing state-of-the-art deep learning methods [8]. Using five CNN models, they used the weighted average of those models to get an accuracy of the 96% in classifying stages of Alzheimer's disease [12]. By removing the last five layers of ResNet50 and adding 8 new layers, achieved 97.2% accuracy in classifying brain tumors and also used Alexnet, Resnet50, Densenet201, InceptionV3, and Googlenet models to classify brain tumors [16].

In some, data are pre-processed in different ways, then models are applied to them. An automated brain disease classification model is created with four main phases, which are preprocessing, exemplar deep feature generator, feature selection, and classification using a support vector machine (SVM) [13]. By using Discrete Cosine Transform-based image fusion, which is combined with a super-resolution and classifier framework, a CNN model, ResNet50, achieved a 98.14% accuracy rate on an open-access dataset [18].

The literature demonstrates a handful of models which have the potential to be used in the brain disease classification sector. However, an in-depth analysis of the most popular and most efficient models on different types of datasets is not quite present in the explored studies.

In our experiment, we have implemented six different models- InceptionV3, ResNet152V2, MobileNetV2, Resnet50, EfficientNetB0, and DenseNet201 on two brain tumor datasets (both individually and manually combined) and an Alzheimer's dataset to visualize the difference of each model, compare their effectiveness by using four different measurements- Accuracy, Precision, Recall, and F1-score, and efficiency.

III. Materials and Methods

A. Dataset

There is a total of three datasets from three different sources, each containing brain MR images of four kinds. Two datasets contain three variants of Brain Tumor, while one dataset contains three variants of Alzheimer's. We have combined two datasets of Brain Tumor to expand the size of the data and reduce biasness. For simplicity purposes, we have named the datasets respectively D1, D2, D3, and D4. D1 represents the Brain Tumor MRI dataset, D2 as Brain Tumor Classification (MRI), D3 as the manual combined dataset, and finally D4 as the Alzheimer's Dataset (4 classes of image). The datasets are publicly available and collected from Kaggle.

In Fig. 1, Fig. 2, and Fig. 3, glioma, meningioma, pituitary, and no tumor are the variations of brain tumor datasets and in Fig. 4, moderate demented, mild demented, non-demented and very mild demented are the four different classes of Alzheimer's dataset. The summary of all datasets is given in Table I.

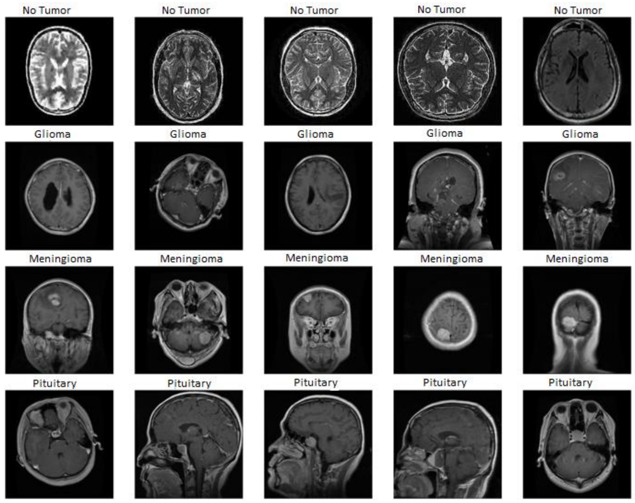

Fig. 1. Sample Images from D1.

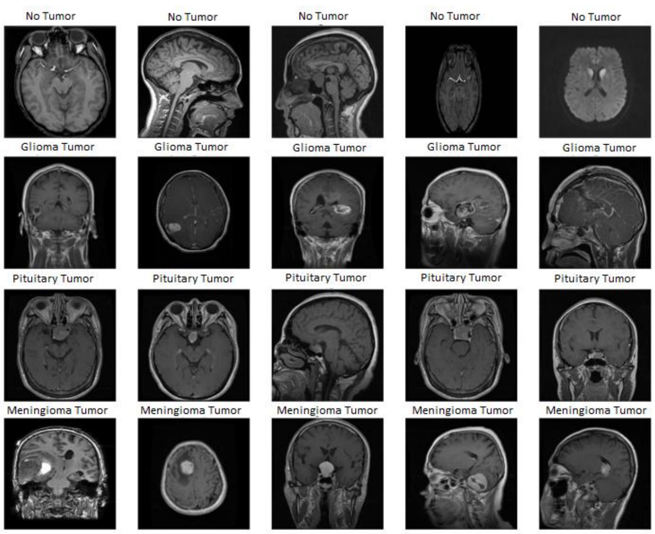

Fig. 2. Sample Images from D2.

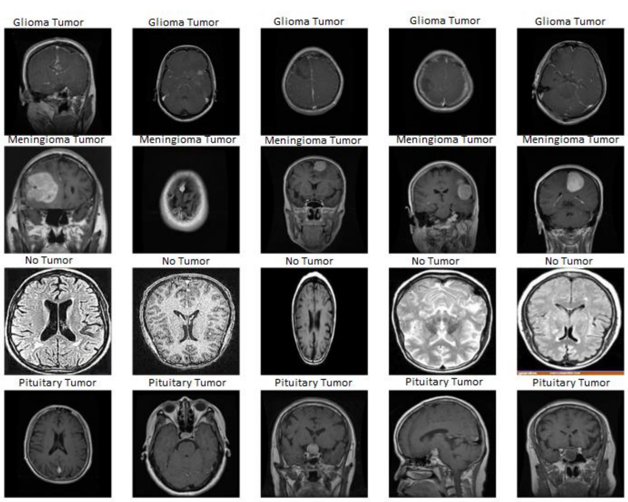

Fig. 3. Sample Images from D3.

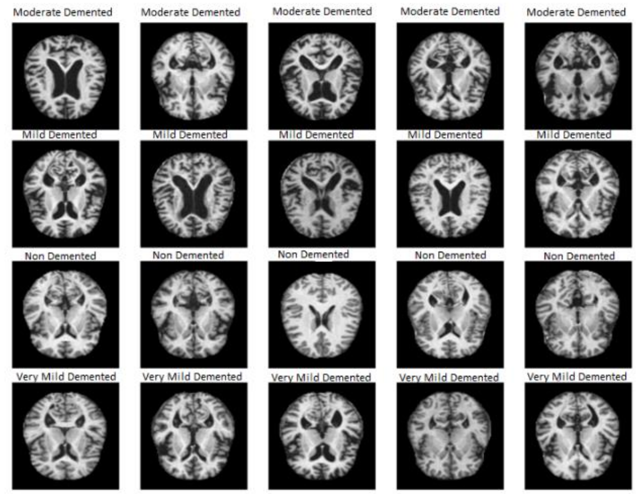

Fig. 4. Sample Images from D4.

TABLE I. SUMMARY OF DATASETS

Dataset	Classes	No. of Images	Total Images
Brain Tumor MRI Dataset (D1)	No Tumor	2000	7023
	Glioma	1621	
	Meningioma	1645	
	Pituitary	1757	
Brain Tumor Classification MRI (D2)	No Tumor	500	3264
	Glioma	926	
	Meningioma	937	
	Pituitary	901	
Manually Combined Dataset (D3)	No Tumor	2500	10000
	Glioma	2500	
	Meningioma	2500	
	Pituitary	2500	
Alzheimer's Dataset (D4)	Moderate Demented	64	6400
	Mild Demented	896	
	Non-Demented	3200	
	Very Mild Demented	2240	

B. Preprocessing

At first, the images are converted into NumPy arrays, where each pixel of an image is assigned to a number, generating an array for each image. Image Augmentation is an important step in image processing. It creates multiple versions of one image to increase the size of the dataset. Each version has different properties that give more information and a new point of view of the image to train. The images are shifted both vertically and horizontally between -20px to 20px, randomly zoomed in and out by 20%. Then the categorical values are converted to numeric values. As there are four classes for each dataset, each class is given a numeric value in the range 0-3.

C. Transfer Learning

Transfer learning is a machine learning technique where a deep learning model reuses the weights that have been generated from a different dataset. The reason for using this method is to use the patterns learned from a similar task to get a head start to avoid huge computational time and attain the best result possible for that model.

The models used in this experiment have already been trained with the ImageNet dataset, which provided us with the weights that we can utilize.

D. Models

Six deep learning models are used train the datasets, these are InceptionV3, ResNet152V2, MobileNetV2, ResNet50, EfficientNetB0, and DenseNet201.

1) InceptionV3: InceptionV3 is the third version of Google's Deep Learning Convolutional Architectures series, Inception. It contains 42-48 layers, which include convolutions, max pooling, average pooling, dropouts, and fully connected layers, and has both symmetric and asymmetric building blocks. This model estimates the marginalized effect of label dropout during training to regularize the classifier layer by changing the label-smoothing regularization (LSR) which is defined by,

$$q'(\mathrm{k}) = (1-\epsilon)\delta_{k,y} + \frac{\epsilon}{k} \quad (1)$$

where the uniform distribution $u(k) = \frac{1}{k}$ is used in the model.

Also by considering the cross entropy, LSR is

$$H(q', p) == (1-\epsilon)H(q,p) + \epsilon H(u,p) \quad (2)$$

LSR prevents the largest logit or unnormalized log probabilities from becoming much larger than all others. It encourages the model to be less confident as it might cause over-fitting and reduce the adapting capability of the model. InceptionV3gave more than 78.1% accuracy on the ImageNet Dataset [3].

2) MobileNetV2: MobileNetV2 has 53 layers, one average pool, and around 350 GFLOPs (Floating point operations per second). It has two types of convolutional layers: 1x1 Convolution and 3x3 Depthwise Convolution. It contains two main blocks, the Inverted Residual Bottleneck Block and Bottleneck Residual Block.

The inverted residual bottleneck layers are implemented in a memory-efficient way to it can be used for mobile applications. It builds a directed acyclic compute hypergraph G, where the edges are the operations and nodes are tensors of intermediate computation. The target is to minimize the total number of tensors stored in memory, so it selects the computation order Σ(G) which has the minimum memory,

$$M(G) = \min_{\pi \in \Sigma(G)} \max_{i \in 1..n} \left[\sum_{A \in R(i,\pi,G)} |A|\right] + size(\pi_i) \quad (3)$$

As for graphs with only trivial parallel structure, the memory needed to compute graph G is

$$\max_{op \in G} \left[\sum_{A \in op_{inp}} |A| + \sum_{B \in op_{out}} |B| + |op|\right] \quad (4)$$

In the bottleneck residual block, a bottleneck block operator $F(x)$can be represented as$F(x) = [A \circ \mathrm{N} \circ \mathrm{B}]$x. As the chain of t tensors of size n/t are the inner tensor, the function can be written as

$$F(x) = \sum_{i=1}^{t} (A_i \circ \mathrm{N} \circ B_i)(x) \quad (5)$$

When t is a small constant between two and five, this method is the most helpful as it can reduce the memory, but can still utilize most of the efficiencies of highly optimized deep learning frameworks [4].

3) ResNet50: ResNet50 is a variant of the Residual Network model. It contains 48 convolutional layers, 1 Max Pool, and 1 Average Pool layer with 3.8 x 10^9 floating point operations per second (FLOPs).

The convolutional layers have 3×3 filters. The layers have the same amount of filters when the output feature map size is the same. However, the layers have the double amount of filters when the feature map size is half. By following these two rules, it performs downsampling. The final layer contains an average pool with a fully-connected layer of 1000 nodes with a softmax function. The total number of weighted layers is 34. Based on this network, shortcut connections are inserted which turn this network into its counterpart residual version. The identity shortcut is defined by

$$y = F(x, \{W_i\}) + x \quad (6)$$

Equation (6) can be directly used when the dimension of x and F are equal. But if the dimensions change, either identity mapping is still performed by the shortcut, or the projection shortcut is used to match dimensions, defined by

$$y = F(x, \{W_i\}) + W_s x \quad (7)$$

For both options, when the shortcuts go across feature maps of two sizes, they are performed with a stride of two. By replacing each two-layer block in the 34-layer net with a three-layer bottleneck block, it results in a 50-layer ResNet [5].

4) EfficientNetB0: EfficientNetB0 is the base model of EfficientNet family. It uses Model Scaling, where the existing model is scaled based on model width, depth, and resolution. This model introduces a new compound scaling method that uniformly scales the width, depth, and resolution to achieve better accuracy using a compound coefficient ∅ . By considering depth, $d = \alpha^{\emptyset}$, width, $w = \beta^{\emptyset}$, and resolution, $r = \gamma^{\emptyset}$, this method is defined as

$$\alpha \cdot \beta^2 \cdot \gamma^2 \approx 2, \text{where } \alpha \geq 1, \beta \geq 1, \gamma \geq 1 \quad (8)$$

For this principle in (8), for any new∅, the total FLOPS will increase by $2^{\emptyset}$ approximately. EfficientNetB0 is scaled up by using the compound scaling method where ∅ is fixed at 1, and after doing a small grid search ofα,β, and γ, the best values are found to be 1.2, 1.1, and 1.15 respectively under the constraint of (8).

The model has a total of 237 layers. It consists of five different modules which are used in a certain way to create each block of the model [6].

5) DenseNet201: DenseNet201 is one of the models of the DenseNet group. It contains 201 layers, and it is divided into Dense Blocks with different filters and the same dimensions for each block. The network includes L(L+1)/2 direct connections. The output of the previous layer becomes the input of the next layer by using composite function operations. The transition layer is added between the Dense Blocks, where it applies batch normalization using downsampling. The growth rate k controls how much information should be added to the next layer. At layer l the growth rate is defined by [7]:

$$k^{[l]} = \left(k^{[0]} + k(l-1)\right) \tag{9}$$

The models are compiled for training by using the compile method of Keras Model Training API. The training data is fit into the model with 10% validation data, 15 epochs, and a batch size of 32. The categorical accuracy of each epoch is monitored to find the highest categorical accuracy by following (10).

$$Categorical\ Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \tag{10}$$

IV. Results and Discussion

A. Result Analysis

We have implemented six algorithms on four different datasets and secured significant results. The maximum accuracy for each dataset was obtained by different models. As we have used various datasets, so the percentage of the accuracy was not consistent in every dataset.

Our models were trained by using 80 percent of the data for training, 10 percent of the training data for validation, and 20 percent for testing. We have used ImageNet as a pre-trained weight in every model and SoftMax for the pre-training classifier. We also included some hyper-parameters as well.

The following figures show how the training and validation accuracy/loss fluctuated per epoch in four different datasets for the models which gave the best results. The red line represents the validation accuracy/loss and the green one is for training accuracy/loss. Also, the final results of each of the models from different datasets are given in Tables II, III, IV, and V. The model with the best accuracy is highlighted in the tables.

TABLE II. Accuracy Summery of D1

Model	Training Accuracy	Validation Accuracy	Testing Accuracy
InceptionV3	**100%**	**97.90%**	**99.54%**
ResNet152V2	100%	96.50%	98.63%
MobileNetV2	99.98%	94.58%	98.09%
ResNet50	100%	98.78%	98.63%
EfficientNetB0	99.98%	99.65%	99.47%
DenseNet201	100%	96.85%	99%

TABLE III. Accuracy Summery of D2

Model	Training Accuracy	Validation Accuracy	Testing Accuracy
InceptionV3	100%	98.26%	76.65%
ResNet152V2	100%	100%	76.90%
MobileNetV2	100%	92.68%	69.04%
ResNet50	100%	99.30%	77.16%
EfficientNetB0	**100%**	**96.17%**	**81.47%**
DenseNet201	100%	97.56%	78.43%

TABLE IV. Accuracy Summery of D3

Model	Training Accuracy	Validation Accuracy	Testing Accuracy
InceptionV3	100%	98.50%	93.45%
ResNet152V2	100%	98.38%	93.10%
MobileNetV2	**99.94%**	**98.75%**	**94.50%**
ResNet50	100%	98.13%	93.45%
EfficientNetB0	100%	99.38%	94.35%
DenseNet201	100%	98.75%	94.10%

TABLE V. Accuracy Summery of D4

Model	Training Accuracy	Validation Accuracy	Testing Accuracy
InceptionV3	100%	99.42%	77.25%
ResNet152V2	100%	98.64%	71.07%
MobileNetV2	100%	100%	76.15%
ResNet50	100%	99.42%	73.50%
EfficientNetB0	**99.98%**	**99.03%**	**78.34%**
DenseNet201	100%	99.61%	75.45%

As we can see different model provides different accuracy based on different datasets. In the dataset D1 shown in Table II, we have achieved 99.54 percent testing accuracy using the InceptionV3 model and the minimum was 98.09 percent using MobileNetV2. In the D2 dataset shown in Table III, we have secured 81.47 percent testing accuracy using the EfficientNetB0 model and 69.04 percent was the minimum by MobileNetV2. After analyzing the second dataset we noticed that there was no equal distribution among the classes, which is why the accuracy might not meet its target. Therefore, we have combined D1 and D2 by maintaining the equal distribution of the four classes and have generated D3. D3 has received improved testing accuracy which is 94.50 percent using MobileNetV2 and ResNet152V2 provided its minimal 93.10 percent shown in Table IV. We have also implemented these models on a different dataset which consisted of the images of Alzheimer's disease and the maximum accuracy has been received at 78.34 percent by EfficientNetB0 shown in Table V. So, we can conclude that InceptionV3, EfficientNetB0, and MobileNetV2 these three models are working best CNN models so far according to our observation.

ResNet models are found to perform comparatively worse than other models. It focuses mainly on creating a deep neural network model without hampering the accuracy. As a result, it takes longer as it has a relatively deep architecture, i.e. more parameters to train. Furthermore, having a deep architecture can also be the reason for its consistent validation loss. In contrast, as we can see in Fig. 5 and Fig. 7 that InceptionV3 and MobileNetV2 take significantly less time for performance. They have divided the convolution layer into two distinct parts. Firstly, instead of separately applying the kernel to all the channels, they have applied depthwise convolution. It mainly applies the kernel to each of the channels individually. Secondly, they applied a convolution with one kernel size to combine the features of the newly generated channel, which directly contributes to the shorter training period. However, as it is apparent from our results, it also sacrifices the overall accuracy. In contrast to MobileNetV2 and InceptionV3, EfficientNet evaluates the scaling part of the neural network. Using a compound coefficient, they uniformly scale the width, resolution, and depth of the network simultaneously to find the best gains. As it is apparent from our results shown in Fig. 6 and Fig. 8, EfficientNetB0 provides significantly better accuracy while taking comparatively less training time. Table VI shows the training and prediction time taken by each model.

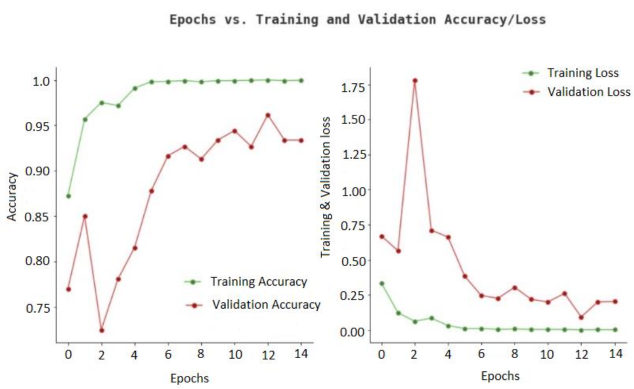

Fig. 5. Epochs vs. Training and Validation Accuracy/Loss of D2 (EfficientNetB0).

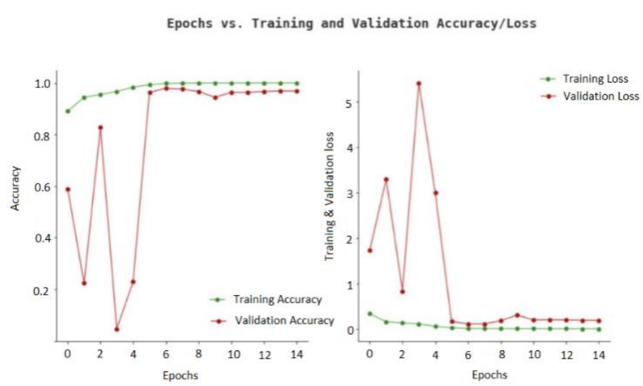

Fig. 6. Epochs vs. Training and Validation Accuracy/Loss of D1 (InceptionV3).

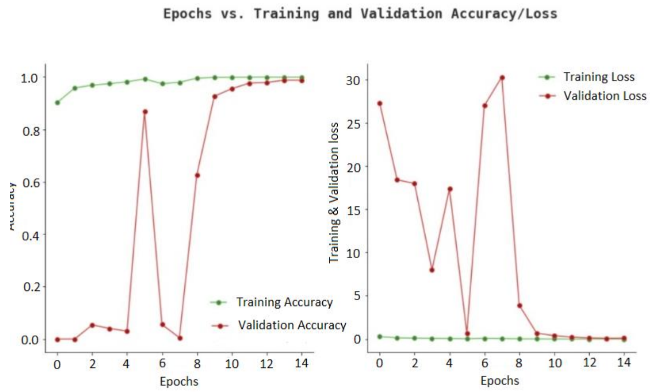

Fig. 7. Epochs vs. Training And Validation Accuracy/Loss of D3 (MobileNetV2).

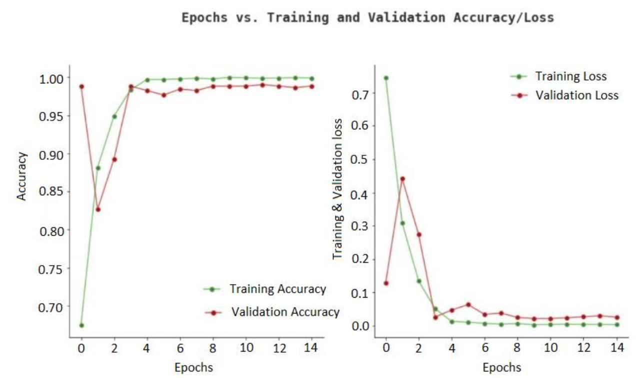

Fig. 8. Epochs vs. Training and Validation Accuracy/Loss of D4 (EfficientNetB0).

TABLE VI. TRAINING AND PREDICTION TIME COMPARISONS

Dataset	Model	Training Time (sec)	Prediction Time (sec)
D1	InceptionV3	~ 1480	~ 8.5
	ResNet152V2	~ 4033 (Worst Case)	~ 12.8
	MobileNetV2	~ 895	~ 4.9
	ResNet50	~ 1706	~ 8.8
	EfficientNetB0	~ 764 (Best Case)	~ 4.2
	DenseNet201	~ 1387	~ 5.6
D2	InceptionV3	~ 778	~ 2.8
	ResNet152V2	~ 925 (Worst Case)	~ 3.6
	MobileNetV2	~ 264 (Best Case)	~ 1.4
	ResNet50	~ 428	~ 4.7
	EfficientNetB0	~ 388	~ 3.1
	DenseNet201	~ 748	~ 2.6
D3	InceptionV3	~ 2573	~ 13.3
	ResNet152V2	~ 5426	~ 15.1
	MobileNetV2	~ 17, 282 (Worst Case)	~ 23.2
	ResNet50	~ 2275	~ 9.8
	EfficientNetB0	~ 1812 (Best Case)	~ 9.4
	DenseNet201	~ 4020	~ 15.7
D4	InceptionV3	~ 591	~ 5.3
	ResNet152V2	~ 1670 (Worst Case)	~ 10.6
	MobileNetV2	~ 457 (Best Case)	~ 2.6
	ResNet50	~ 703	~ 4.6
	EfficientNetB0	~ 670	~ 3.8
	DenseNet201	~ 1276	~ 6.3

As we know validation of these models is very important to prove our performance for this research purpose, we have used some metrics to do so- precision, recall, f1 score, and confusion matrix. The mathematical notations of these terms are given below:

$$Precision = \frac{true\ positive}{true\ positive + false\ positive} \quad (11)$$

$$Recall = \frac{true\ positive}{true\ positive + false\ negative} \quad (12)$$

$$F1\ Score = 2 \times \frac{precision \times recall}{presion + recall} \quad (13)$$

The confusion matrices in Fig. 9 to 12, and the evaluation results calculated by (11), (12), and (13) in Tables VII, VIII, IX, and X of the models with the best results from each dataset are shown below:

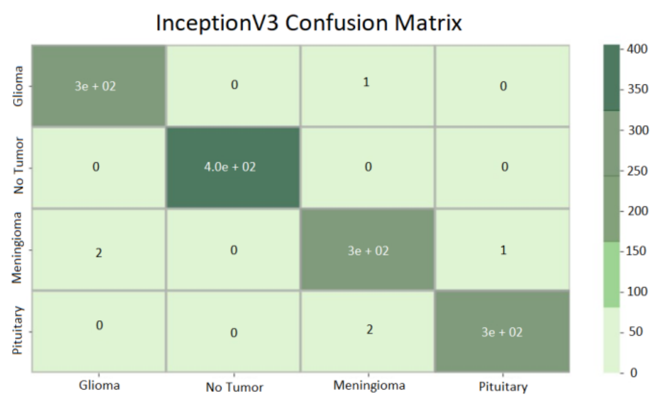

Fig. 9. Confusion Matrix of D1.

TABLE VII. EVALUATION RESULT OF D1 (INCEPTIONV3)

Class	Precision	Recall	F1-Score	Accuracy
Glioma	0.99	1.00	1.00	1.00
Meningioma	1.00	1.00	1,00	
Pituitary	0.99	0.99	0.99	
No Tumor	1.00	0.99	0.99	

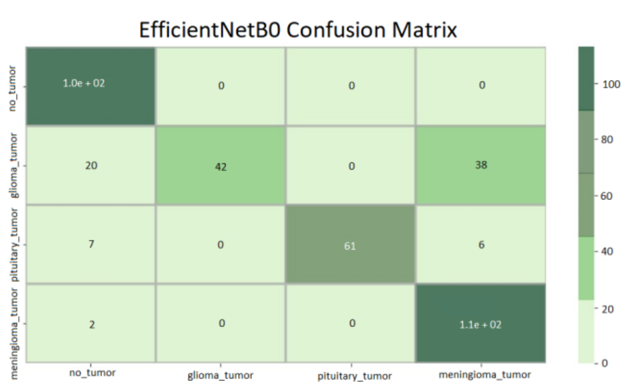

Fig. 10. Confusion Matrix of D2.

TABLE VIII. EVALUATION RESULT OF D2 (EFFICIENTNETB0)

Class	Precision	Recall	F1-Score	Accuracy
Glioma	0.78	1.00	0.88	0.81
Meningioma	1.00	0.42	0.59	
Pituitary	1.00	0.82	0.90	
No Tumor	0.72	0.98	0.83	

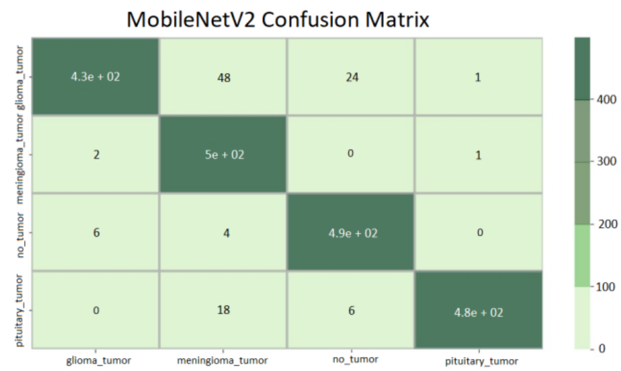

Fig. 11. Confusion Matrix of D3.

TABLE IX. EVALUATION RESULT OF D3 (MOBILENETV2)

Class	Precision	Recall	F1-Score	Accuracy
Glioma	0.98	0.85	0.91	0.94
Meningioma	0.88	0.99	0.93	
Pituitary	0.94	0.98	0.96	
No Tumor	1.00	0.95	0.97	

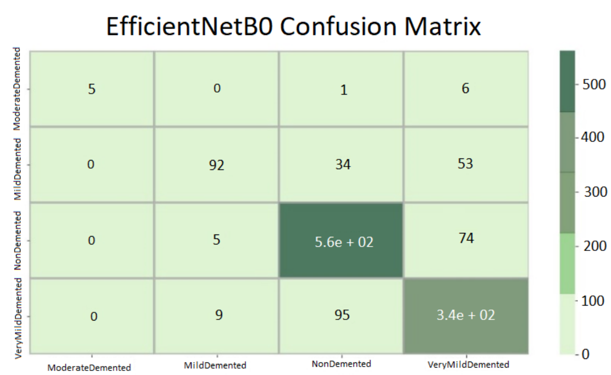

Fig. 12. Confusion Matrix of D4.

TABLE X. EVALUATION RESULT OF D4 (EFFICIENTNETB0)

Class	Precision	Recall	F1-Score	Accuracy
Moderate Demented	1.00	0.42	0.59	0.78
Mild Demented	0.87	0.51	0.65	
Non Demented	0.81	0.88	0.84	
Very Mild Demented	0.72	0.77	0.74	

B. Performance

For comparison purposes, we have mentioned different types of models and their accuracy percentages in the relevant sector.

TABLE XI. COMPARISON TABLE WITH OTHER MODELS

Serial	Model	Dataset Size	Accuracy (%)
1.	DCT-CNN-ResNet50 [18]	70,220	98.14%
2.	ELM-LRF [25]	220,875	97.18%
3.	Multiscale Convolutional Neural Network [23]	3,264	97.30%
4.	Multiclass SVM cubic classifier [21]	335	99.8%
5.	InceptionV3 [10]	411	85%
6.	ResNet152V2 [11]	7,023	98.90%
7.	MobileNetV2 [15]	2,475	94%
8.	EfficientNetB0 [19]	6,400	92.98%
9.	InceptionV3 (Our model)	7,023	99.54%
10.	EfficientNetB0 (Our model)	3,264	81.47%
11.	MobileNetV2 (Our model)	10,000	94.50%
12.	EfficientNetB0 (Our model)	6,400	78.34%

From Table XI, we can see many other studies have used CNN models as well. In [11], the author found the highest accuracy of 98.90% by the ResNet152V2 model, where we have managed to use IncpetionV3 to achieve 99.54% using the same dataset. However, our EfficientNetB0 model did not outperform the model used in [23] and [19], where both datasets were identical. Although for other models all of us did not use the same dataset, we cannot compare the accuracy for every model (e.g., [26]) entirely.

C. Challenges

The study uses deep learning models, which is a very time-consuming procedure even with the help of transfer learning. The amount of data was not enough, generating some difficulties while getting a better result. Also, the use of medical datasets introduced its' own challenges, because this sector has an extreme restriction on the time limit and the results have to be monitored as accurately as possible, as the application of these models in actual medical settings is expected in the future.

V. CONCLUSION

Our goal was to find which CNN model or models can provide the maximum result. That is why we have implemented six models in four datasets of different types. Our datasets were not large enough to study in this field and not all datasets are suitable for every algorithm, which resulted in different accuracy in different datasets with the same algorithm. As we are using medical data, it is challenging to predict exact results, yet we have secured a remarkable accuracy of 99.54 percent. In medical science, time is one of the most crucial factors for any emergency. Therefore, by implementing the most promising CNN models, we can be able to analyze brain MRI images straight away. Mostly in our country, doctors try to identify the types of tumors manually as well as any other brain diseases, so the risk of the occurrence of human error is high. Sometimes patients would have to use high-priced diagnostical methods to find the types of the disease. Health-related issues are way too sensitive, hence any type of mistake is unacceptable. Therefore, this study could save time and cost, and most importantly could save lives.

By using the most prominent model, we can develop software that can be used to detect a tumor or any other disease in an instant. We have only worked with brain MRI images, in the future, we would like to work with different types of MRI images as well.

ACKNOWLEDGMENT

We are grateful to Dr. Bidyut Kumar Saha, MBBS, MD (India), Post Doctoral Training in Neuro Cardio Radiology (AIIMS, Delhi), Senior Consultant & Coordinator – Diagnostic & Interventional Radiology, Asgar Ali Hospital. He guided and collaborated with us and shared his medical knowledge regarding our dataset analyzing part. Without his help, we would not be able to do our tasks flawlessly.

REFERENCES

[1] Z. Akkus, A. Galimzianova, A. Hoogi, D. L. Rubin, and B. J. Erickson, "Deep Learning for Brain MRI Segmentation: State of the Art and Future Directions," J. Digit. Imaging, vol. 30, no. 4, pp. 449–459, 2017, doi: 10.1007/s10278-017-9983-4.

[2] I. Despotović, B. Goossens, and W. Philips, "MRI segmentation of the human brain: Challenges, methods, and applications," Comput. Math. Methods Med., vol. 2015, 2015, doi: 10.1155/2015/450341.

[3] C. Szegedy, V. Vanhoucke, S. Ioffe, and J. Shlens, "Rethinking the Inception Architecture for Computer Vision."

[4] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., pp. 4510–4520, 2018, doi: 10.1109/CVPR.2018.00474.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," Accessed: Feb. 05, 2022. [Online]. Available: http://image-net.org/challenges/LSVRC/2015/.

[6] M. Tan and Q. V Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks."

[7] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," Accessed: Feb. 05, 2022. [Online]. Available: https://github.com/liuzhuang13/DenseNet.

[8] N. Noreen, S. Palaniappan, A. Qayyum, I. Ahmad, M. Imran, and M. Shoaib, "A Deep Learning Model Based on Concatenation Approach for the Diagnosis of Brain Tumor," IEEE Access, vol. 8, pp. 55135–55144, 2020, doi: 10.1109/ACCESS.2020.2978629.

[9] S. S. Yadav and S. M. Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis," J. Big Data, vol. 6, no. 1, pp. 1–18, Dec. 2019, doi: 10.1186/S40537-019-0276-2/TABLES/16.

[10] A. Kumar, P. Pathak, and P. Stynes, "A Transfer Learning Approach to Classify the Brain Age from MRI Images," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 12581 LNCS, pp. 103–112, Dec. 2020, doi: 10.1007/978-3-030-66665-1_8.

[11] A. Alnemer and J. Rasheed, "An Efficient Transfer Learning-based Model for Classification of Brain Tumor," ISMSIT 2021 - 5th Int. Symp. Multidiscip. Stud. Innov. Technol. Proc., pp. 478–482, 2021, doi: 10.1109/ISMSIT52890.2021.9604677.

[12] S. U. Sadat, H. H. Shomee, A. Awwal, S. N. Amin, M. T. Reza, and M. Z. Parvez, "Alzheimer's Disease Detection and Classification using Transfer Learning Technique and Ensemble on Convolutional Neural

Networks," pp. 1478–1481, Jan. 2022, doi: 10.1109/SMC52423.2021.9659179.

[13] A. Kursad Poyraz, S. Dogan, E. Akbal, and T. Tuncer, "Automated brain disease classification using exemplar deep features," Biomed. Signal Process. Control, vol. 73, p. 103448, Mar. 2022, doi: 10.1016/J.BSPC.2021.103448.

[14] Y. K. Cetinoglu, I. O. Koska, M. E. Uluc, and M. F. Gelal, "Detection and vascular territorial classification of stroke on diffusion-weighted MRI by deep learning," Eur. J. Radiol., vol. 145, p. 110050, Dec. 2021, doi: 10.1016/J.EJRAD.2021.110050.

[15] T. H. Arfan, M. Hayaty, and A. Hadinegoro, "Classification of Brain Tumours Types Based on MRI Images Using Mobilenet," 2021 2nd Int. Conf. Innov. Creat. Inf. Technol. ICITech 2021, pp. 69–73, Sep. 2021, doi: 10.1109/ICITECH50181.2021.9590183.

[16] A. Çinar and M. Yildirim, "Detection of tumors on brain MRI images using the hybrid convolutional neural network architecture," Med. Hypotheses, vol. 139, p. 109684, Jun. 2020, doi: 10.1016/J.MEHY.2020.109684.

[17] L. V. Fulton, D. Dolezel, J. Harrop, Y. Yan, and C. P. Fulton, "Classification of Alzheimer's Disease with and without Imagery Using Gradient Boosted Machines and ResNet-50," Brain Sci. 2019, Vol. 9, Page 212, vol. 9, no. 9, p. 212, Aug. 2019, doi: 10.3390/BRAINSCI9090212.

[18] A. Deshpande, V. V. Estrela, and P. Patavardhan, "The DCT-CNN-ResNet50 architecture to classify brain tumors with super-resolution, convolutional neural network, and the ResNet50," Neurosci. Informatics, vol. 1, no. 4, p. 100013, Dec. 2021, doi: 10.1016/J.NEURI.2021.100013.

[19] S. Savaş, "Detecting the Stages of Alzheimer's Disease with Pre-trained Deep Learning Architectures," Arab. J. Sci. Eng. 2021, pp. 1–18, Sep. 2021, doi: 10.1007/S13369-021-06131-3.

[20] H. A. Khan, W. Jue, M. Mushtaq, and M. U. Mushtaq, "Brain tumor classification in MRI image using convolutional neural network," Math. Biosci. Eng., vol. 17, no. 5, pp. 6203–6216, 2020, doi: 10.3934/MBE.2020328.

[21] M. I. Sharif, M. A. Khan, M. Alhussein, K. Aurangzeb, and M. Raza, "A decision support system for multimodal brain tumor classification using deep learning," Complex Intell. Syst., no. 0123456789, 2021, doi: 10.1007/s40747-021-00321-0.

[22] A. M. Alqudah, H. Alquraan, I. A. Qasmieh, A. Alqudah, and W. Al-Sharu, "Brain tumor classification using deep learning technique - A comparison between cropped, uncropped, and segmented lesion images with different sizes," Int. J. Adv. Trends Comput. Sci. Eng., vol. 8, no. 6, pp. 3684–3691, 2019, doi: 10.30534/ijatcse/2019/155862019.

[23] F. J. Díaz-Pernas, M. Martínez-Zarzuela, D. González-Ortega, and M. Antón-Rodríguez, "A deep learning approach for brain tumor classification and segmentation using a multiscale convolutional neural network," Healthc., vol. 9, no. 2, 2021, doi: 10.3390/healthcare9020153.

[24] J. Seetha and S. S. Raja, "Brain tumor classification using Convolutional Neural Networks," Biomed. Pharmacol. J., vol. 11, no. 3, pp. 1457–1461, 2018, doi: 10.13005/bpj/1511.

[25] A. Ari and D. Hanbay, "Deep learning based brain tumor classification and detection system," Turkish J. Electr. Eng. Comput. Sci., vol. 26, no. 5, pp. 2275–2286, 2018, doi: 10.3906/elk-1801-8.

[26] M. F. Siddiqui, G. Mujtaba, A. W. Reza, and L. Shuib, "Multi-class disease classification in brain MRIs using a computer-aided diagnostic system," Symmetry, vol. 9, no. 3, pp. 1–14, 2017, doi: 10.3390/sym9030037.