|  |  |  |
| --- | --- | --- |
|  | Engineering Science and Technology, an International Journal  journal homepage: www.elsevier.com/locate/jestch | C:\Users\0002402\Desktop\Jestch.jpg |

Brain Tumor Classification using Deep Learning Techniques with Magnetic Resonance Imaging

Sumya Aktera,\*, Shah Mainur Rahmana and Maruf Kabir Refata

aDepartment of Computer Science and Engineering, Bangladesh Army University of Science and Technology

|  |  |  |
| --- | --- | --- |
| ARTICLE INFO |  | ABSTRACT |
| Article history:  Received  Received in revised form  Accepted  Available online |  | The classification of brain tumor is a significant issue in computer-aided diagnosis (CAD). A usually utilized medical imaging modality is Magnetic Resonance Imaging (MRI), which helps to find abnormal cells or tissues, which are referred to as tumors. But it is hard for clinicians to study the MRI data to find or diagnose brain tumors as the shapes and locations of tumors are different. To overcome this limitation, researchers have made many efforts in previous years. This paper focuses on the analysis of the performance of various deep learning techniques. In the pre-processing step we split the dataset, cropped the images and resized all the images to train transfer learning based CNN models. Then we performed data augmentation to increase the size of the dataset. In this research, we introduced 16 pre-trained convolutional neural network(CNN) architectures: VGG16, Resnet50, InceptionV3, Xception, InceptionResNetV2, MobileNet, DenseNet121, NasNetMobile, EfficientNets(B0-B7). Then we analysed the results to figure out which algorithm will perform better. Experiment findings reveal that ResNet50 performs better than the other CNN architectures with 96% accuracy and 96.77% AUC score.  2009 Elsevier Ltd. All rights reserved. |
| Keywords:  Brain Tumor  Deep Learning  Transfer Learning  Convolutional Neural Networks (CNN)  Image Processing |  |
|  |  |  |

1. Introduction

The brain is the most essential organ of the body, which regulates all the body functions. When the brain is harmed, it will affect multiple facets of life, including memories, sensations, and even personality. There are several different kinds of brain disorders, and one of them is brain tumor.[[1]](#footnote-2)\*

A brain tumor occurs when there is an uncontrolled and abnormal development of tissues in the brain. Two types of brain tumors exist. One of them is benign, which has a low rate of growth and does not contain cancer cells. Another is malignant tumors, which contain cancer cells, and have a fast progression rate. Malignant tumors are invasive in nature[1,2]. A patient can survive if the tumor is detected early, but it is challenging to measure the size and resolution of the tumor with the naked eye[3].

The use of Magnetic Resonance Imaging(MRI) in the diagnosis and treatment of brain tumors is important. An MRI scan can locate aneurysms and tumors in the brain, as well as distinguish between white and grey matter. Where frequent imaging is required for diagnosis or treatment, particularly in the brain, MRI is the modality of choice, since it does not use x-rays or other forms of radiation[4]. MRI images give better outcomes than CT scan images, and therefore neurosurgeons prefer MRI scans[5,6].

Tumor classification has been performed manually in earlier days. In order to reduce the time of diagnosis and overcome any mistakes, before making any decisions, automated tumor detection and classification methods are required[7]. Over the last couple of years, this procedure has become easier with less or no human involvement due to the advent of various machine learning and deep learning methods[8,9].

Techniques for machine learning allow systems to learn from experience. Machine learning refers to the ability of a system to gain and incorporate information through large-scale observations and to expand and extend itself instead of being programmed with that knowledge through learning new knowledge[10]. It is a subset of artificial intelligence(AI) approaches aimed to develop algorithms that learn the principles of interpretation from training samples, and apply them to new data from the same domain in order to make informed decisions[11].

Deep learning has been shown to outperform the machine learning methods for a specific class problem, which allows models to work so well or better than humans[12,13]. There are various deep learning techniques for the classification of brain tumors. In this research, we used deep learning CNN architectures.

1. Related works

Identify and classify the brain tumor from MRI is a challenging task. Deep learning approaches have recently proved to be useful for clinical research and have found themselves in applications such as identifying brain tumors. There are multiple studies to identify and classify tumors automatically. For the classification of the two classes of brain MR images, a transfer learning-based technique was used by Priyansh Saxena et al. [14]. Here, pre-trained Inception-V3, ResNet-50, VGG-16 were applied. The accuracy of ResNet-50 was 95% and F1 score was also 95%, which was the highest among those models. L. Jagjeevan Rao et al. [5] applied KNN, BPNN(Backpropagation Neural Network), and CNN for the classification of brain tumor. CNN performs better compared to other classifiers with 91% accuracy, 92% specificity and 91% sensitivity. Asmita Dixit et al. [15] proposed the impact of accuracy in classification by various pre-processing techniques, which were checked on various kernel SVMs, to classify malignant and benign tumor. Their highest accuracy was 61.53%, using pre-processing technique. A brain tumor image classification system, which classify the tumors into three subtypes(Glioma, Pituitary, Meningioma) using SVM and CNN was proposed by Shubham Kumar Baranwal et al. [16], where CNN performed better than SVM. For separate classes, the AUC value of CNN was Glioma 94%, Pituitary 99% and Meningioma 93%. V. Bhanumathi et al. [3] classified brain tumor using pre-trained AlexNet, VGG-16 and GoogleNet. In their experiment GoogleNet showed higher accuracy than AlexNet and VGG-16. Fatemeh BashirGonbadi et al.[17] proposed a CNN architecture to diagnose and classify glioma brain tumors and achieved 99.18% accuracy. Mrs. A. Sankari et al. [18] proposed a system for brain tumor segmentation using CNN with 3x3 filter size and after performing image enhancement, they have achieved 90% accuracy. Yakub Bhanothu et al. [7] proposed deep learning Faster R-CNN algorithm to detect and classify 3 types of tumor, namely glioma, meningioma and pituitary. The proposed approach uses the VGG-16 model as a base layer of the classifier network and the regional proposal network. Their proposed algorithm obtained a mean average accuracy of 77.60% for all the groups as a performance measure. A simple artificial neural network(ANN) method was proposed by Muhammad Nazir et al. [19] to classify tumors by normal and abnormal brain. There are three processing stages of their proposed model and those are pre-processing, extraction of features and MR images classification into normal and abnormal. The overall accuracy of their proposed method was 91.8%. N. Jagan Mohana Rao et al. [20] proposed an automatic segmentation method using CNN and they have achieved 91.3% accuracy.

There are some relevant works which are mentioned in table 1.

**Table 1.**Relevant works

|  |  |  |
| --- | --- | --- |
| Ref. | Dataset | Classification Technique |
| [3] | 50 sample brain MRIs (20 normal, 30 abnormal) | AlexNet, VGG 16, GoogleNet |
| [5] | 200 sample brain MRIs (100 tumorous, 100 normal) | KNN, BPNN, CNN, and a mixture of mean filtering and fuzzy c means |
| [7] | Jun Cheng et al. [21,22] | Faster R-CNN, that uses VGG-16 as base network |
| [14] | Kaggle Dataset. 253 sample brain MRIs (155 tumorous, 98 normal) | VGG 16, Inception V3 , ResNet 50 |
| [16] | Nanfang Hospital and General Hospital, Tianjin Medical University (3064 MRI images) | CNN, SVM |
| [18] | IBSR and BRATS 2015 | CNN |
| [20] | BRATS 2015, 2013 and from [23] | CNN |
| [24] | BRATS 2017 | FCN, DenseCRF, K-means |
| [25] | 700 brain MRI images (500 for training, 200 for validation) | CNN |
| [26] | Kaggle dataset | CNN |
| [27] | 1892 sample MRIs (1666 for training, 226 for testing) | CNN with Softmax, Radial Basis Function, Decision Tree |
| [28] | BRATS | CNN, SVM, MLP, KNN, Logistic Regression, Random Forest, Naïve Bayes |
| [29] | BRATS 2017 | PixelNet |
| [30] | BRATS 2018 | Ensemble of CNN |
| [31] | Figshare (3064 MRI images) | CNN, SVM, KNN |
| [32] | REMBRANDT , BRAINS, University of Edinburgh ImageBank repository and from MIRIAD | CNN |
| [33] | Govt. Medical College, Calicut, India[34] | GoogleNet, AlexNet, InceptionV3, ResNet 50 |
| [35] | BRATS 2015 | CNN |
| [36] | Kaggle dataset | BrainMRNet |
| [37] | 100 MRI images for training and 40 MRI images for testing | AlexNet |
| [38] | BRATS 2013 | CNN, PNN |
| [39] | BRATS 2013 | CNN |

1. Deep Learning and Convolutional Neural Networks

In radiology, the sub-set of machine learning known as deep learning has recently become a hot topic. Deep learning has proven to outperform other machine learning approaches for a particular class of problems, enabling models to be built that perform as well or even better than human ones [12,13].

One of the most popular Deep Learning methods is CNN, in the medical imaging field[40,41]. Basic convolutional neural network architecture includes input layer, hidden layers and output layer. CNN hidden layers includes: convolutional layers, pooling layers, fully connected layers and normalization layers[42,43]. CNN is a model of deep learning, learns like the brains of humans. On CNN-predicted algorithms, research also shows good performance. CNNs are able to solve a lot of object apperception as well as biological image segmentation challenges[20,35].



Fig. 1. VGG-16 architecture

* + 1. VGG-16

VGG-16 is a 13-layer convolutional neural network with three fully connected layers[44]. There are other layers, such as max pool layer, that do not contain any trainable parameters.

This model’s input size is 224x224x3. The VGG-16 architecture consists of two contiguous blocks of two convolutional layers each, followed by max-pooling. After that, there are three contiguous blocks of three convolutional layers, followed by max-pooling. Finally, it has three layers that are totally linked.

* + 1. ResNet-50

The concept of skip connection is used by ResNet-50. We can resolve the problem vanishing gradients using skip connection[45].



Fig.2. Skip connection

Convolutional layers are staked one after the other in figure 2, adding the initial input to the convolution block’s output. This method is known as skip connection.

Fig. 3. Identity block

The network learns from the difference between the input (X) and output (Y) and make it Y=X. In Y=F(X), if F(X) is 0 then the input will be equal to output.

Y = X+F(X)

Y = X+0

Y = X

When the network’s input and output is both the same, the identity block is used.

Fig. 4. Convolutional block

Convolutional block is used when the input of the network is not same as the output of the network. We can add the X value into F(X) if the input size and the output size are same. To solve this problem there are 2 ways for matching the output size. One is padding the input volume and the other is performing 1\*1 convolutions.

The input image of ResNet-50 is 224x224x3 in size. There are three convolution layers in each convolution block, and three convolution layers in each identity block. Then this input goes to the 7x7 convolution which filter is 64 with stride 2. Then, max pooling is performed with 3x3 window size and stride 2. After that, identity block is used in the next three blocks, which means input size and output size of the image is same. In the next block convolutional block is used to make the size of input and output same.

Fig. 5. ResNet-50 architecture

* + 1. Inception-V3

With a low equipped device, Inception-V3 is difficult to train directly and it takes at least a few days to train it[46]. We used transfer learning here to preserve the parameters of the previous layer and delete the last layer of Inception-V3. Then we retrain the last layer. As a result, it reduces the computational time and increases model’s accuracy.

Fig. 6. Inception V3 architecture

In the architecture of Inception-V3 model, the input size is 299x299x3. Then 3 convolutional layers are used where filter size is 3x3. The first convolution layer contains 32 stride 2 filters, the second convolution layer contains 32 stride 1 filters, and the third convolution layer contains 64 stride 1 filters. After third convolutional layer the image size will be 147x147x64. Then there is a max pooling layer where window size is 3x3 and stride is 2. After performing max pooling, the image size will be 73x73x64. Next there is a convolution layer, which has 80 filters of size 1x1 and the image size will be 73x73x80. Then in the next step, the convolution layer has 192 filters of size 3x3 and the image size will be 71x71x192. After that, max pool layer has 3x3 window size with stride 2. After running inception block A 3 times, Reduction block A is performed. In reduction block A, parameters are reduced. Next, inception block B runs 4 times. Then it goes to reduction block B. In the next step inception block C runs 2 times. Then there is an average pool layer and 2 fully connected layers. In between inception block B and reduction block B, there is also a classifier. In the final layer there are 1000 neurons because the algorithm is trained on 1000 different classes. The auxiliary classifiers are used before softmax classifiers to solve the vanishing gradient problem.

* + 1. Xception

Xception is a convolutional neural network which is 71 layers deep. It has the same number of parameters as Inception-V3. Xception is based on the fact that the spatial correlation is entirely separate from the correlation between the input channels[47,48].

The model is separated into 3 flows namely, Entry flow, Middle flow and Exit flow. In the entry flow the input size is 299x299x3. The convolution layers are followed by ReLU activation in the entry flow. Xception model is processed by Separable convolutional layers and positioned in the entire framework of deep learning. Max pool layers are used. Also, there are skip connections in this architecture. At the end of the entry flow the size of feature map is 19x19x728. In the middle flow, feature map is taken as input. The block of middle flow is repeated 8 times. In exit flow Max pooling and Global Average pooling is performed. 2048 dimensional vectors are connected with the optimal fully connected layers in the exit flow.

* + 1. Inception-ResNet V2

Inception-ResNet V2 is 164 layers deep. The input size of this network is 299x299. Inception-ResNet V2 adds residual connections to the input that add the output of the inception module’s convolution operation[49]. For the residual extension to function, the input and output after convolution must have the same dimensions. Therefore, we use 1x1 convolutions to balance the depth sizes after the original convolutions.



Fig. 7. Inception-ResNet A block

In figure 7, Inception-ResNet A block is shown. The Inception-ResNet V2 network has a schema for the 35x35 grid module. Here, ReLU activations are used and convolution operations are performed. The residual connections were used to replace the pooling operation within the main inception modules. In the reduction blocks, we can find those operations.

There is a schema for the 17x17 grid module of Inception-ResNet V2 network in Inception-ResNet B. Here 7x7 convolutions are factorized and ReLU activations are used.

There is a schema for the 8x8 grid module of the Inception-ResNet V2 network in Inception-ResNet C. Here, the factorization process happens in an asymmetric way.



Fig. 8. Inception-ResNet B block



Fig. 9. Inception-ResNet C block



Fig. 10. Inception-ResNet V2 architecture

Inception-ResNet V2 is a network in the style of Inception, but instead of filter concatenation, it uses residual connection. The input image is taken first, followed by the stem. Next, the Inception-ResNet A block is executed 5 times and the output goes to the Reduction A block. Next, Inception-ResNet B block is performed 10 times and after that, the output goes to Reduction B block. Then, Inception-ResNet C is executed 5 times and average pooling is performed. The Inception-ResNet V2 is a hybrid Inception variant that improves recognition efficiency dramatically.

* + 1. MobileNet

MobileNet is a CNN architecture that was developed for real-world mobile applications in mind. MobileNets mainly use depthwise separable convolutions instead of the regular convolutions used in previous architectures to construct lighter models[50]. The width multiplier and resolution multiplier are two new global hyperparameters introduced by MobileNets that enable model developers to trade off latency, speed accuracy, and low size based on their needs. Inception models are also introduced by MobileNet to minimize computation in the first few layers. To detect brain tumors MobileNet architecture is used based on transfer learning. The default input size of this network is 224x224.

Fig. 11. Depthwise Convolution

The channel-wise DKxDK spatial convolution is known as depthwise convolution. If we have six channels in figure 11, we would have six DKxDK spatial convolutions. It’s a different map for each input channel of a single convolution. As a result, the number of output channels equals the number of input channels. Df2 \* M \* Dk2 is its computational cost.

Fig. 12. Pointwise Convolution

Since depthwise convolution only filters the input channel, no new features are created. As a consequence, a pointwise convolution layer is applied. The 1x1 convolution is used to adjust the dimension of pointwise convolution. M \* N \* Df2 is the computational cost.

Inputs are filtered and combined into a new output set by standard convolution in one step. Batch normalization[51] and with the exception of the final fully connected layer, which has no nonlinearity and feeds into a softmax layer for classification, ReLU nonlinearity is applied to all layers of the MobileNet architecture.

Fig. 13. Standard Convolution

The width multiplier is a global hyperparameter used to create smaller and less costly models. Width multiplier is denoted by α and the value is between 0 and 1.

The second parameter that is added in MobileNet is called the multiplier of the resolution. This hyperparameter is used to minimize to input image’s resolution, and then decreases the input to each layer with the same factor.

* + 1. DenseNet 121

DenseNet stands for Densely Connected Convolutional Network, which is one of the latest discoveries for visual object identification in neural networks. It is a CNN architecture somewhat similar to that of ResNet with certain basic variations. DenseNet concatenates the output of the previous layer with the next layer. On some image classification benchmarks, it is more effective.

Fig. 14. Blocks used in the model

Here the input image is 224x224x3 in size. The whole model is divided into 2 parts or blocks, which are dense block and transition block[52]. DenseNet has a feature layer that captures low-level features from images, as well as many dense blocks and transition layers between them. There are 121 layers in DenseNet 121. The structure has a variety of advantages over current structures, including solving the vanishing gradient problem, optimizing feature propagation, promoting feature reuse, and decreases the amount of parameters[53].

At the end of the network, the input of shape 224x224x3 is down sampled to 7x7x512. In figure 14(a) several dense layers make up a dense block. Inside the dense blocks, the feature map size remains the same. To minimize computation, a 1x1 convolutional layer with 128 filters is inserted and reduces size of feature map, which ensures that the second convolutional layer maintains a constant input depth[54]. The input volume and the product of the two operations are concatenated in the action of adding new information to the general knowledge of the network.

To prevent this problem, the features in figure 14(b) are reduced and abstracted after a dense block with a limited number of dense layers. A batch normalization layer, 1x1 convolution, and a 2x2 avg pooling layer with stride 2 (to minimize the size to half) are used in the DenseNet architecture. The depth is reduced to a fixed number by the 1x1 convolution layer, while the size is reduced by average pooling.

In figure 14(c), the output of figure 14(b) is taken as input. Here 2x2 transpose convolution with stride 2 is performed.

* + 1. NasNet Mobile

NASNet Mobile is the abbreviation for Neural Architecture Search Mobile. NasNet mobile is NasNet’s mobile version[55]. The NasNet architecture is made up of simple building blocks, which are optimized by reinforcement learning. A cell is made up of just a few operations that are repeated several times depending on the network’s capacity requirements. NasNet mobile is the mobile version of NasNet with 564 million multiply-accumulates, 12 cells, and 5.3 million parameters.

* + 1. EfficientNet

EfficientNet is a CNN architecture and scaling method that uses a compound coefficient to uniformly scale all depth, distance, and resolution parameters[56]. Unlike standard practice, which scales parameters at random, the EfficientNet scaling technique uses a series of predetermined scaling coefficients to systematically scale the network’s distance, depth, and resolution. EfficientNet scales network distance, depth, and resolution in a principled fashion using a compound coefficient.

* 1. Transfer Learning

Transfer learning is a way to reuse pre-trained model information for another task. In order to allow a model to attain greater performance in the domain of interest, the core concept of transfer learning is borrowing labelled data or information derived from other similar domains. In the problem of classification, regression and clustering transfer learning can be used[44]. Since our dataset is small to train CNN, we used transfer learning. The CNN models of this paper are pre-trained on ImageNet dataset. The training data contains 1000 classes and 1.2 million images[57]. figure 15 shows the concept of transfer learning.

Fig. 15. Transfer Learning

In this experiment, we have used transfer learning method for various deep learning CNNs which are, VGG16, ResNet50, InceptionV3, Xception, InceptionResNetV2, MobileNet, DenseNet121, NasNetMobile, EfficientNet-B0 to EfficintNet-B7.

1. Methodology

The aim of this experiment is to identify and classify brain tumor from MR images. Different types of deep learning approaches are applied on MRI images to classify the normal and abnormal brain. First, we collected a dataset. Then data pre-processing is done on the collected dataset. After that, we performed data augmentation. Finally, classification is done using various pre-trained CNN architectures.

Fig. 16. Methodology Overview

* 1. Data Collection

We collected the dataset from Kaggle[58] containing two folders of images, one of which is for negative samples and another is for positive samples. There are total of 253 images, from which 155 MRI images with tumor and 98 normal brain MRI images.

* 1. Image Preprocessing

To correctly identify the tumor from MRI images preprocessing is essential part. To make the images efficient for training preprocessing plays an important role. We have implemented three main steps for image preprocessing. Splitting the dataset is the first step, cropping the images is the second step and resizing the image is the third step.

* + 1. Data Splitting

In this stage dataset splits into three sections: Train, Test and Validation. The training set refers to the data sample which is used to fit the model. Our model will be trained on the same data in our training set over and over again during each epoch and the model will continue learning about features of the dataset. 76% of our entire dataset is training data. To validate the model, validation set is used. 20% of entire dataset is validation data. To test the model we used test set, which is 4% of our entire dataset.

* + 1. Crop Images

To crop the MR images, we performed certain steps [14,59]. First of all, to blur the images a bit we used Gaussian filter. Gaussian blur tends to reduce high-frequency noise and blurs the uninterested areas, allowing one to focus on the fundamental MR image structure. Then we performed image thresholding, which allows the brain field to be segmented from the image. Moreover, we performed morphological operations, which are erosion and dilation to remove small regions of noise. Dilation removes the object noise completely and the noise of the background can be minimized by performing erosion. Then we find contours in the thresholded images and select the largest one. After that, we find north, south, east and west four extreme points. Finally, we cropped the images which contain only the brain. Figure 17 shows one sample image, that were cropped.



Fig. 17. Crop the image which contains only the brain

* + 1. Resize images

Images in various dimensions and aspect ratios are used in the input dataset. Each network’s input is different in size. So, we resized the images to perform deep learning CNN architectures. Table 2 shows the required input sizes for 16 CNN models.

**Table 2.**Input shape for 16 CNN architectures

|  |  |
| --- | --- |
| Model | Input Shape |
| VGG-16 | 224x224 |
| ResNet-50 | 224x224 |
| Inception-V3 | 299x299 |
| Xception | 299x299 |
| InceptionResNet-V2 | 299x299 |
| MobileNet | 224x224 |
| DenseNet-121 | 224x224 |
| NasNetMobile | 224x224 |
| EfficientNet-B0 | 224x224 |
| EfficientNet-B1 | 240x240 |
| EfficientNet-B2 | 260x260 |
| EfficientNet-B3 | 300x300 |
| EfficientNet-B4 | 380x380 |
| EfficientNet-B5 | 456x456 |
| EfficientNet-B6 | 528x528 |
| EfficientNet-B7 | 600x600 |



* 1. Data Augmentation

Approaches of data augmentation include a range of strategies, which extend artificially the dataset size to train the models for deep learning CNNs. Since our dataset size is small, we performed data augmentation. It is done by rotation of the image, shearing, flipping, changing the brightness, rescale etc. 10 augmented image samples are shown in figure 18. The increase in data has attracted considerable interest in applications of deep learning, particularly following the emergence of deep convolutional neural networks [40,45,60,61].

Fig. 18. Data Augmentation

**Table 3.**Performance of 16 CNN architectures

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Model | Precision | Recall | F1-score | Kappa coefficient (k) | AUC-ROC Score | Training Time (sec) | Accuracy |
| VGG-16 | 96.46% | 87.09% | 91.52% | 79.40% | 90.91% | 5411.17 | 90% |
| ResNet-50 | **100%** | 93.54% | **96.66%** | **91.68%** | **96.77%** | 1847.22 | **96%** |
| Inception-V3 | 66.90% | 93.54% | 77.33% | 16.83% | 57.30% | 2386.62 | 66% |
| Xception | 71.79% | 90.32% | 80% | 35.30% | 66.21% | 4096.51 | 72% |
| InceptionResNet-V2 | 65.71% | 74.19% | 69.69% | 11.50% | 55.51% | 5799.46 | 60% |
| MobileNet | 79.49% | **100%** | 88.57% | 63.03% | 78.94% | **518.72** | 84% |
| DenseNet-121 | 77.78% | 90.32% | 83.58% | 50.80% | 74.11% | 1512.19 | 78% |
| NasNetMobile | 67.44% | 93.55% | 78.38% | 22.63% | 59.93% | 1155.91 | 68% |
| EfficientNet-B0 | 85.71% | 96.77% | 90.91% | 73.45% | 85.23% | 700.57 | 88% |
| EfficientNet-B1 | 88.23% | 96.77% | 92.30% | 78.10% | 87.86% | 1156.14 | 90% |
| EfficientNet-B2 | 85.71% | 96.77% | 90.31% | 73.45% | 85.22% | 1691.11 | 88% |
| EfficientNet-B3 | 93.75% | 96.77% | 95.23% | 87.13% | 93.12% | 2754.13 | 94% |
| EfficientNet-B4 | 85.29% | 93.54% | 89.23% | 69.35% | 83.61% | 5363.89 | 86% |
| EfficientNet-B5 | 90.32% | 90.32% | 90.32% | 74.53% | 87.26% | 12070.91 | 88% |
| EfficientNet-B6 | 83.33% | 96.77% | 89.55% | 68.69% | 82.59% | 23545.88 | 86% |
| EfficientNet-B7 | 88.57% | **100%** | 93.93% | 82.30% | 89.47% | 39819.49 | 92% |

* 1. Results and Analysis

For the experiment we used Google Colab to implement CNN architectures. We trained each CNN models for 50 epochs. Table 1 shows the performance of 16 deep learning CNN architectures used in this experiment. In terms of the F1 score, Cohen’s kappa(k), accuracy and AUC-ROC score, the performance of models were calculated.

|  |  |  |
| --- | --- | --- |
|  | Positive | Negative |
| True | TP | TN |
| False | FP | FN |

Confusion Matrix is used for the evaluation of performance. Confusion matrix compares the model’s predictions to the actual target values. True Positive(TP), True Negative(TN), False Positive(FP) and False Negative(FN) are all possible outcomes[20,62]. If the classification is done correctly then it is either True Positive(TP), or True Negative(TN). On the other hand, if the classification is incorrect then it is either False Positive(FP), or False Negative(FN).

The percentage of positive identifications, which are correct, is called precision.

Precision = TP/(TP+FP)

The percentage of actual positives which is identified correctly is called recall.

Recall = TP/(TP+FN)

F1 Score = 2\*(Precision\*Recall) / (Precision+Recall)

AUC-ROC Score = Measurement of Area under the ROC curve.

The efficiency of effective classification to the total number of classification tests is measured by accuracy of classification[63,64].

Accuracy = (TP+TN) / (TP+TN+FP+FN)

In figure 19, analysis based on the accuracy of classifiers is shown.

Due to higher precision and recall values, among all other deep learning CNN approaches implemented in this experiment, ResNet-50 had the highest F1 score which is 96.66%. The accuracy, AUC-ROC score is highest for ResNet-50, which is respectively 96% and 96.77%. If False Negative rate is high, then the model will predict a patient doesn’t have a brain tumor, who actually has a tumor. In ResNet-50 the False Negative rate is zero on test data. Also, ResNet-50 has the highest kappa coefficient(k) and F1 score, which is respectively 91.68% and 96.66%. By analysis and comparison of performance we can see, ResNet-50 has performed better than all other CNN architecture which were implemented in our experiment.

For other CNN architectures if the network goes deeper, then it is difficult to choose parameters that learn the identity function and that’s why the performance decreases. But identity function is easy to learn for the residual block in ResNet for skip connection. Therefore, adding extra layers doesn’t decrease the performance of ResNet.

Fig. 19. Analysis based on the accuracy of classifiers

1. Conclusion

Detecting the presence of brain tumor from MRI in a fast and accurate manner is a challenging problem. In this study, performance of various deep learning techniques was analyzed to identify and classify the brain tumor with. 16 pre-trained deep learning CNN architectures were used in this experiment. By analyzing and comparing the results we found that ResNet-50 performs better than any other models in this experiment with highest accuracy and AUC-ROC score which is respectively 96% and 96.77%. In future we can further develop the experiment by identifying the actual size of tumor and improve the CNN model performance by tuning transfer learned model.

**References**

[1] K. Munir, F. Frezza, A. Rizzi, Deep Learning for Brain Tumor Segmentation, in: U. Kose, J. Alzubi (Eds.), Deep Learn. Cancer Diagnosis, Springer Singapore, Singapore, 2021: pp. 189–201. https://doi.org/10.1007/978-981-15-6321-8\_11.

[2] S. Dube, S. El-saden, T.F. Cloughesy, U. Sinha, Content Based Image Retrieval for MR Image Studies of Brain Tumors, (2006).

[3] V. Bhanumathi, R. Sangeetha, CNN Based Training and Classification of MRI Brain Images, 2019 5th Int. Conf. Adv. Comput. Commun. Syst. (2019) 129–133.

[4] R. Prasad, N.P.G. Seshadri, R. Periyasamy, S. Miller, A. Bit, K. Mitra, Electroencephalography and near-infrared spectroscopy-based hybrid biomarker for brain imaging, Elsevier Ltd, 2019. https://doi.org/10.1016/B978-0-08-102420-1.00009-1.

[5] L.J. Rao, R. Challa, D. Sudarsa, C. Naresh, C.Z. Basha, Enhanced automatic classification of brain tumours with FCM and convolution neural network, Proc. 3rd Int. Conf. Smart Syst. Inven. Technol. ICSSIT 2020. (2020) 1233–1237. https://doi.org/10.1109/ICSSIT48917.2020.9214199.

[6] D. Deodhare, N.N.R. Ranga, S.R. Amit, Preprocessing and Image Enhancement Algorithms for a Form-based Intelligent Character Recognition System, Int. J. Comput. Sci. Appl. 2 (2005) 131–144.

[7] Y. Bhanothu, A. Kamalakannan, G. Rajamanickam, Detection and Classification of Brain Tumor in MRI Images using Deep Convolutional Network, 2020 6th Int. Conf. Adv. Comput. Commun. Syst. ICACCS 2020. (2020) 248–252. https://doi.org/10.1109/ICACCS48705.2020.9074375.

[8] V. Makde, J. Bhavsar, S. Jain, P. Sharma, Deep neural network based classification of tumourous and non-tumorous medical images, Smart Innov. Syst. Technol. 84 (2018) 199–206. https://doi.org/10.1007/978-3-319-63645-0\_22.

[9] A.S. Lundervold, A. Lundervold, An overview of deep learning in medical imaging focusing on MRI, Z. Med. Phys. 29 (2019) 102–127. https://doi.org/10.1016/j.zemedi.2018.11.002.

[10] S. Balasubramanyam, Machine Learning, IFPUG Guid. to IT Softw. Meas. (2012) 385–386. https://doi.org/10.1201/b11884-28.

[11] G. Chassagnon, M. Vakalopolou, N. Paragios, M.P. Revel, Deep learning: definition and perspectives for thoracic imaging, Eur. Radiol. 30 (2020) 2021–2030. https://doi.org/10.1007/s00330-019-06564-3.

[12] E.J. Hwang, S. Park, K.N. Jin, J.I. Kim, S.Y. Choi, J.H. Lee, J.M. Goo, J. Aum, J.J. Yim, J.G. Cohen, G.R. Ferretti, C.M. Park, Development and Validation of a Deep Learning-Based Automated Detection Algorithm for Major Thoracic Diseases on Chest Radiographs, JAMA Netw. Open. 2 (2019) e191095. https://doi.org/10.1001/jamanetworkopen.2019.1095.

[13] D. Ardila, A.P. Kiraly, S. Bharadwaj, B. Choi, J.J. Reicher, L. Peng, D. Tse, M. Etemadi, W. Ye, G. Corrado, D.P. Naidich, S. Shetty, End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography, Nat. Med. 25 (2019) 954–961. https://doi.org/10.1038/s41591-019-0447-x.

[14] P. Saxena, A. Maheshwari, S. Maheshwari, Predictive Modeling of Brain Tumor: A Deep Learning Approach, in: M.K. Sharma, V.S. Dhaka, T. Perumal, N. Dey, J.M.R.S. Tavares (Eds.), Innov. Comput. Intell. Comput. Vis., Springer Singapore, Singapore, 2021: pp. 275–285.

[15] A. Dixit, A. Nanda, MR brain image tumor classification via kernel SVM with different preprocessing techniques, Springer Singapore, 2019. https://doi.org/10.1007/978-981-13-9939-8\_20.

[16] S.K. Baranwal, K. Jaiswal, K. Vaibhav, A. Kumar, R. Srikantaswamy, Performance analysis of Brain Tumour Image Classification using CNN and SVM, Proc. 2nd Int. Conf. Inven. Res. Comput. Appl. ICIRCA 2020. (2020) 537–542. https://doi.org/10.1109/ICIRCA48905.2020.9183023.

[17] F.B. Gonbadi, H. Khotanlou, Glioma brain tumors diagnosis and classification in MR images based on convolutional neural networks, 2019 9th Int. Conf. Comput. Knowl. Eng. ICCKE 2019. (2019) 375–379. https://doi.org/10.1109/ICCKE48569.2019.8965143.

[18] A. Sankari, S. Vigneshwari, Automatic tumor segmentation using convolutional neural networks, ICONSTEM 2017 - Proc. 3rd IEEE Int. Conf. Sci. Technol. Eng. Manag. 2018-Janua (2017) 268–272. https://doi.org/10.1109/ICONSTEM.2017.8261291.

[19] M. Nazir, F. Wahid, S. Ali Khan, A simple and intelligent approach for brain MRI classification, J. Intell. Fuzzy Syst. 28 (2015) 1127–1135. https://doi.org/10.3233/IFS-141396.

[20] N. Jagan Mohana Rao, B. Anil Kumar, Brain Tumour Identification Through MRI Images Using Convolution Neural Networks, Springer International Publishing, 2020. https://doi.org/10.1007/978-3-030-24643-3\_125.

[21] J. Cheng, W. Huang, S. Cao, R. Yang, W. Yang, Z. Yun, Z. Wang, Q. Feng, Enhanced performance of brain tumor classification via tumor region augmentation and partition, PLoS One. 10 (2015) 144479. https://doi.org/10.1371/journal.pone.0140381.

[22] J. Cheng, W. Yang, M. Huang, W. Huang, J. Jiang, Y. Zhou, R. Yang, J. Zhao, Y. Feng, Q. Feng, W. Chen, Retrieval of Brain Tumors by Adaptive Spatial Pooling and Fisher Vector Representation, PLoS One. 11 (2016) 1–15. https://doi.org/10.1371/journal.pone.0157112.

[23] B.H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, L. Lanczi, E. Gerstner, M.A. Weber, T. Arbel, B.B. Avants, N. Ayache, P. Buendia, D.L. Collins, N. Cordier, J.J. Corso, A. Criminisi, T. Das, H. Delingette, Ç. Demiralp, C.R. Durst, M. Dojat, S. Doyle, J. Festa, F. Forbes, E. Geremia, B. Glocker, P. Golland, X. Guo, A. Hamamci, K.M. Iftekharuddin, R. Jena, N.M. John, E. Konukoglu, D. Lashkari, J.A. Mariz, R. Meier, S. Pereira, D. Precup, S.J. Price, T.R. Raviv, S.M.S. Reza, M. Ryan, D. Sarikaya, L. Schwartz, H.C. Shin, J. Shotton, C.A. Silva, N. Sousa, N.K. Subbanna, G. Szekely, T.J. Taylor, O.M. Thomas, N.J. Tustison, G. Unal, F. Vasseur, M. Wintermark, D.H. Ye, L. Zhao, B. Zhao, D. Zikic, M. Prastawa, M. Reyes, K. Van Leemput, The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS), IEEE Trans. Med. Imaging. 34 (2015) 1993–2024. https://doi.org/10.1109/TMI.2014.2377694.

[24] L. Yang, W. Jiang, H. Ji, Z. Zhao, X. Zhu, A. Hou, Automatic brain tumor segmentation using cascaded FCN with denseCRF and K-means, 2019 IEEE/CIC Int. Conf. Commun. China, ICCC 2019. (2019) 545–549. https://doi.org/10.1109/ICCChina.2019.8855974.

[25] N. Abiwinanda, M. Hanif, S.T. Hesaputra, A. Handayani, T.R. Mengko, Brain tumor classification using Convolutional Neural Networks, Springer Singapore, 2018. http://dx.doi.org/10.1007/978-981-10-9035-6\_33.

[26] C.L. Choudhury, C. Mahanty, R. Kumar, B.K. Mishra, Brain Tumor Detection and Classification Using Convolutional Neural Network and Deep Neural Network, 2020 Int. Conf. Comput. Sci. Eng. Appl. ICCSEA 2020. (2020) 0–3. https://doi.org/10.1109/ICCSEA49143.2020.9132874.

[27] M. Siar, M. Teshnehlab, Brain tumor detection using deep neural network and machine learning algorithm, 2019 9th Int. Conf. Comput. Knowl. Eng. ICCKE 2019. (2019) 363–368. https://doi.org/10.1109/ICCKE48569.2019.8964846.

[28] T. Hossain, F.S. Shishir, M. Ashraf, M.A. Al Nasim, F. Muhammad Shah, Brain Tumor Detection Using Convolutional Neural Network, 1st Int. Conf. Adv. Sci. Eng. Robot. Technol. 2019, ICASERT 2019. 2019 (2019) 1–6. https://doi.org/10.1109/ICASERT.2019.8934561.

[29] M. Islam, H. Ren, Multi-modal PixelNet for brain tumor segmentation, Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 10670 LNCS (2018) 298–308. https://doi.org/10.1007/978-3-319-75238-9\_26.

[30] A. Kori, M. Soni, B. Pranjal, M. Khened, V. Alex, G. Krishnamurthi, Ensemble of Fully Convolutional Neural Network for Brain Tumor Segmentation from Magnetic Resonance Images, in: A. Crimi, S. Bakas, H. Kuijf, F. Keyvan, M. Reyes, T. van Walsum (Eds.), Brainlesion Glioma, Mult. Sclerosis, Stroke Trauma. Brain Inj., Springer International Publishing, Cham, 2019: pp. 485–496.

[31] S. Deepak, P.M. Ameer, Brain tumor classification using deep CNN features via transfer learning, Comput. Biol. Med. 111 (2019) 103345. https://doi.org/10.1016/j.compbiomed.2019.103345.

[32] N.M. Balasooriya, R.D. Nawarathna, A sophisticated convolutional neural network model for brain tumor classification, 2017 IEEE Int. Conf. Ind. Inf. Syst. ICIIS 2017 - Proc. 2018-Janua (2018) 1–5. https://doi.org/10.1109/ICIINFS.2017.8300364.

[33] A.P. Rahmathunneesa, K. V. Ahammed Muneer, Performance analysis of pre-trained deep learning networks for brain tumor categorization, Proc. 2019 9th Int. Conf. Adv. Comput. Commun. ICACC 2019. (2019) 253–257. https://doi.org/10.1109/ICACC48162.2019.8986151.

[34] Government Medical College Kozhikode, (n.d.). https://www.govtmedicalcollegekozhikode.ac.in/ (accessed March 18, 2021).

[35] S. Pereira, A. Pinto, V. Alves, C.A. Silva, Brain Tumor Segmentation Using Convolutional Neural Networks in MRI Images, 35 (2016) 1240–1251.

[36] M. Toğaçar, B. Ergen, Z. Cömert, BrainMRNet: Brain tumor detection using magnetic resonance images with a novel convolutional neural network model, Med. Hypotheses. 134 (2020) 109531. https://doi.org/10.1016/j.mehy.2019.109531.

[37] P. Muthu Krishnammal, S. Selvakumar Raja, Deep learning based image classification and abnormalities analysis of MRI brain images, Proc. 2019 TEQIP - III Spons. Int. Conf. Microw. Integr. Circuits, Photonics Wirel. Networks, IMICPW 2019. (2019) 427–431. https://doi.org/10.1109/IMICPW.2019.8933239.

[38] G. Madhupriya, M. Guru Narayanan, S. Praveen, B. Nivetha, Brain tumor segmentation with deep learning technique, Proc. Int. Conf. Trends Electron. Informatics, ICOEI 2019. 2019-April (2019) 758–763. https://doi.org/10.1109/icoei.2019.8862575.

[39] R. Lang, L. Zhao, K. Jia, Brain tumor image segmentation based on convolution neural network, Proc. - 2016 9th Int. Congr. Image Signal Process. Biomed. Eng. Informatics, CISP-BMEI 2016. (2017) 1402–1406. https://doi.org/10.1109/CISP-BMEI.2016.7852936.

[40] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, in: F. Pereira, C.J.C. Burges, L. Bottou, K.Q. Weinberger (Eds.), Adv. Neural Inf. Process. Syst., Curran Associates, Inc., 2012. https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[41] A. Shalbaf, M. Vafaeezadeh, Automated detection of COVID-19 using ensemble of transfer learning with deep convolutional neural network based on CT scans, Int. J. Comput. Assist. Radiol. Surg. (2020). https://doi.org/10.1007/s11548-020-02286-w.

[42] T. Kalaiselvi, Deriving tumor detection models using convolutional neural networks from MRI of human brain scans, Int. J. Inf. Technol. (2020) 2–7. https://doi.org/10.1007/s41870-020-00438-4.

[43] M. Havaei, A. Davy, D. Warde-farley, A. Biard, A. Courville, Y. Bengio, C. Pal, P. Jodoin, H. Larochelle, Brain tumor segmentation with Deep Neural Networks, Med. Image Anal. 35 (2017) 18–31. https://doi.org/10.1016/j.media.2016.05.004.

[44] S. Tammina, Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images, 9 (2019) 143–150. https://doi.org/10.29322/IJSRP.9.10.2019.p9420.

[45] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2016-Decem (2016) 770–778. https://doi.org/10.1109/CVPR.2016.90.

[46] X. Xia, C. Xu, B. Nan, Inception-v3 for flower classification, 2017 2nd Int. Conf. Image, Vis. Comput. ICIVC 2017. (2017) 783–787. https://doi.org/10.1109/ICIVC.2017.7984661.

[47] F. Chollet, Xception: Deep learning with depthwise separable convolutions, Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017. 2017-Janua (2017) 1800–1807. https://doi.org/10.1109/CVPR.2017.195.

[48] T. Xiao, L. Liu, K. Li, W. Qin, S. Yu, Z. Li, Comparison of Transferred Deep Neural Networks in Ultrasonic Breast Masses Discrimination, Biomed Res. Int. 2018 (2018). https://doi.org/10.1155/2018/4605191.

[49] C. Szegedy, S. Ioffe, V. Vanhoucke, A.A. Alemi, Inception-v4, inception-ResNet and the impact of residual connections on learning, 31st AAAI Conf. Artif. Intell. AAAI 2017. (2017) 4278–4284.

[50] A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, MobileNets: Efficient convolutional neural networks for mobile vision applications, ArXiv. (2017).

[51] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, 32nd Int. Conf. Mach. Learn. ICML 2015. 1 (2015) 448–456.

[52] M. Shaikh, G. Anand, G. Acharya, A. Amrutkar, V. Alex, G. Krishnamurthi, Brain tumor segmentation using dense fully convolutional neural network, Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 10670 LNCS (2018) 309–319. https://doi.org/10.1007/978-3-319-75238-9\_27.

[53] Y. Zhou, Z. Li, H. Zhu, C. Chen, M. Gao, K. Xu, J. Xu, Holistic brain tumor screening and classification based on densenet and recurrent neural network, Springer International Publishing, 2019. https://doi.org/10.1007/978-3-030-11723-8\_21.

[54] The Efficiency of Densenet. From Restnets, Highway networks, and… | by Hao Gao | Medium, (n.d.). https://medium.com/@smallfishbigsea/densenet-2b0889854a92 (accessed March 16, 2021).

[55] B. Zoph, V. Vasudevan, J. Shlens, Q. V. Le, Learning Transferable Architectures for Scalable Image Recognition, Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (2018) 8697–8710. https://doi.org/10.1109/CVPR.2018.00907.

[56] M. Tan, Q. V. Le, EfficientNet: Rethinking model scaling for convolutional neural networks, 36th Int. Conf. Mach. Learn. ICML 2019. 2019-June (2019) 10691–10700.

[57] ImageNet, (n.d.). http://www.image-net.org/about-stats (accessed February 19, 2021).

[58] Brain MRI Images for Brain Tumor Detection | Kaggle, (n.d.). https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection (accessed February 17, 2021).

[59] Finding extreme points in contours with OpenCV - PyImageSearch, (n.d.). https://www.pyimagesearch.com/2016/04/11/finding-extreme-points-in-contours-with-opencv/ (accessed February 17, 2021).

[60] B. Abdollahi, N. Tomita, S. Hassanpour, Data Augmentation in Training Deep Learning Models for Medical Image Analysis, in: L. Nanni, S. Brahnam, R. Brattin, S. Ghidoni, L.C. Jain (Eds.), Deep Learn. Deep Learn. Descriptors Med. Appl., Springer International Publishing, Cham, 2020: pp. 167–180. https://doi.org/10.1007/978-3-030-42750-4\_6.

[61] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc. (2015) 1–14.

[62] J. Naik, P.S. Patel, Tumor Detection and Classification Using Decision Tree in Brain MRI, Int. J. Eng. Dev. Res. 14 (2013) 49–53.

[63] N. Varuna Shree, T.N.R. Kumar, Identification and classification of brain tumor MRI images with feature extraction using DWT and probabilistic neural network, Brain Informatics. 5 (2018) 23–30. https://doi.org/10.1007/s40708-017-0075-5.

[64] G. Madhikar, P.S.S. Lokhande, Brain Tumour Detection and Classification by Using Modified Region Growing Method : A Review, 2 (2013) 2316–2320.

1. \* Corresponding author.

   E-mail address: [sumya.hstu@gmail.com](mailto:sumya.hstu@gmail.com) (Sumya Akter), [mainur.cse@gmail.com](mailto:mainur.cse@gmail.com) (Shah Mainur Rahman), [marufrefat3666@gmail.com](mailto:marufrefat3666@gmail.com) (Maruf Kabir Refat) [↑](#footnote-ref-2)