Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/MBE

MBE, 17(5): 6203–6216.
DOI: 10.3934/mbe.2020328
Received: 08 July 2020
Accepted: 31 August 2020
Published: 15 September 2020

Research article

Brain tumor classification in MRI image using convolutional neural network

Hassan Ali Khan[1], **Wu Jue**[1,*], **Muhammad Mushtaq**[2] **and Muhammad Umer Mushtaq**[3,4]

[1] School of Computer Science and Technology, Southwest Unversity of Science and Techonlogy, Mianyang 621010, China

[2] Insitute for Neuro and Bioinformatics, University of Lübeck, Germany

[3] School of Information Engineering, Southwest Unversity of Science and Techonlogy, Mianyang 621010, China

[4] Department of Software Engineering, MUST, Mirpur AJK, Pakistan

* **Correspondence:** Email: wujue@aliyun.com; Tel: +8613808113612.

Abstract: Brain tumor is a severe cancer disease caused by uncontrollable and abnormal partitioning of cells. Recent progress in the field of deep learning has helped the health industry in Medical Imaging for Medical Diagnostic of many diseases. For Visual learning and Image Recognition, task CNN is the most prevalent and commonly used machine learning algorithm. Similarly, in our paper, we introduce the convolutional neural network (CNN) approach along with Data Augmentation and Image Processing to categorize brain MRI scan images into cancerous and non-cancerous. Using the transfer learning approach we compared the performance of our scratched CNN model with pre-trained VGG-16, ResNet-50, and Inception-v3 models. As the experiment is tested on a very small dataset but the experimental result shows that our model accuracy result is very effective and have very low complexity rate by achieving 100% accuracy, while VGG-16 achieved 96%, ResNet-50 achieved 89% and Inception-V3 achieved 75% accuracy. Our model requires very less computational power and has much better accuracy results as compared to other pre-trained models.

Keywords: brain tumor; MRI; deep learning; CNN; transfer learning; VGG; inception; resnet

1. Introduction

A brain is the most sensitive organ of our body, which controls the core functions and characteristics in the human body and according to the National Brain Tumor Society, in the United States, about 700,000 people live with a brain tumor, and the figure will rise to 787,000 by the end of 2020 [1]. Compared with other cancers such as breast cancer or lung cancer, a brain tumor is not more common but, still, a brain tumor is the number 10th leading cause of deaths worldwide. An estimated 18,020 adults will die this year from brain cancer [2]. Brain tumor has a lasting and psychological impact on patient life.The Brain tumor is caused by tissue abnormality that develops within the brain or in the central spine, interrupting proper brain function. A Brain tumor is marked as Benign and Malignant. Benign brain tumors do not contain cancer cells and grow gradually. They do not spread and commonly stay in one region of the brain, whereas malignant brain tumors contain cancer cells and grow quickly and spread through to other brain and spine regions as well. A Malignant tumor is life-threatening and harmful.World Health Organization (WHO) has graded brain tumors according to brain health behavior, into grade 1 and 2 tumors that are low-grade tumors also known as benign tumors, or grade 3 and 4 tumors which are high-grade tumors also known as malignant tumors [3]. The Brain tumor is diagnosed using several techniques such as CT scan, EEG, but Magnetic Resource Image (MRI) is the most effective and widely used method. MRI uses powerful and effective magnetic fields and radio waves to generate internal images of the organs within the body. MRI provides more detailed information on the internal organs and is, therefore, more effective than CT or EEG scanning.

In the past few years because of AI and Deep learning, significant advancement has been made in the medical science like Medical Image processing technique which helps doctors to the diagnose disease early and easily, before that, it was tedious and time-consuming. So to resolve such kind of limitations computer-aided technology is much needed because Medical Field needs efficient and reliable techniques to diagnose life-threatening diseases like cancer, which is the leading cause of mortality globally for patients. So in our study with the help of Brain MRI Images, we provide a method for classification of brain tumors into cancerous and non-cancerous using data augmentation technique and convolutional neural network model.

2. Related works

Artificial intelligence and deep learning are primarily used in image processing techniques to segment, identify, and classify MRI Images and are also used to classify and detect brain tumors. So many works have already been done on the classification and segmentation of brain MRI images. Some of the international journals we reviewed on the detection and classification of brain tumor using deep learning are Sheikh Basheera et al., [4] proposed a method for classifying brain tumors where the tumor is initially segmented from an MRI image and segmented portion is then extracted through a pre-trained convolutional neural network using stochastic gradient descent. Muhammad Sajjad et al. [5] suggested classification of multi-grade tumors by applying data augmentation technique to Mri images and then tuning it using a pre-trained VGG-19 CNN Model. Carlo, Ricciardi et al., [6] presented an approach for classifying pituitary adenomas tumor MRIs by using multinomial logistic regression and k-nearest neighbor algorithms. The approach achieved an accuracy of 83% on multinomial logistic regression and 92% on a k-nearest neighbor with an AUC curve of 98.4%. Khwaldeh, saed et al., [7] presented a framework for classification of brain MRI images into healthy and unhealthy, and a grading system for categorizing unhealthy brain images into low and high grades, by modifying the Alex-Net CNN model which revealed 91% accuracy. Nyoman Abiniwanda et al., [8] trained a convolutional neural network to classify three specific brain tumors classes, namely Meningioma, Glioma, and Pituitary, which achieved 98.51% training accuracy and 84.19% validation accuracy. Sunanda Das et al., [9] also trained a CNN model with an image processing technique to identify various brain tumor types and achieved 94.39% accuracy with an average precision of 93.33%. Romeo, Valeria et al., [10] presented a radiomic machine learning approach to predict tumor grades and nodal status from CT scans of primary tumor lesions and got the highest accuracy of 92.9% by Naive Bayes and k-nearest neighbor. Muhammed Talo et al., [11] used the ResNet34 pre-trained CNN model a transfer learning approach along with Data Augmentation to classify normal and abnormal brain MRI images and got 100% accuracy. Arshia Rehman et al., [12] used three different pre-trained CNN models (VGG16, AlexNet, and GoogleNet) to classify the brain tumors into pituitary, glioma, and meningioma. During this Transfer learning approach, VGG16 acquires the highest accuracy that is 98.67%. Ahmet inar et al., [13] modified the pre-trained ResNet50 CNN model by removing its last 5 layers and adding 8 new layers instead and comparing its accuracy with other pre-trained models such as GoogleNet, AlexNet, ResNet50. The updated ResNet50 model showed effective results by achieving 97.2% accuracy.

The unavailability of labeled data is one of the major obstacles in the penetration of deep learning in medical healthcare. As recent development of deep learning applications in other fields has shown that the bigger the data would be the better accuracy result will be. Data segmentation and data augmentation are done using deep learning in the mentioned literature, and different pre-trained CNN Models using the transfer learning approach to classify brain tumors had been used. Most of the literature addresses the classification efficiency using transfer learning approach. The pre-trained models that are mostly used in the mentioned literature are VGG-16, ResNet-50 and Inception-v3, which are pre-trained on a mass amount of datasets such as ImageNet. And for radiology research and experiments, we have to do fine-tuning by freezing the layers to reduce parameters if the dataset is small, we also have to replace the fully connected layers according to the dataset labels, Besides transfer learning requires high processing power from specialized processors (GPUs) to train smoothly, which is cost consuming, and one of another drawback in transfer learning is that the image input size is fixed so, we have to adjust our images according to the pre-trained model's input size. so in our experiment, we took a very small dataset of Brain MRI Images. We applied the data augmentation technique along with the image processing technique on those MRI images and then trained a CNN model from scratch on that augmented preprocessed image data to determine whether the MRI image contains a tumor or not. And at last, we compared the diagnostic performance and computational consumption of our model with the VGG-16 and ResNet Model.

3. Proposed methods

In this research, we applied Image Processing and Data Augmentation techniques on a small dataset of 253 brain MRI images [14]. We trained them through a simple 8 Convolutional layers CNN model and compared our scratched CNN model accuracy with pre-trained VGG-16, ResNet-50, and Inception-v3 models using transfer learning approach. The dataset includes 155 images of malignant cancer and 98 of benign non-cancerous tumors. We split our dataset into 3 separate segments for training, validation, and testing. The training data is for model learning, validation data is sample data for model evaluation and model parameters tuning. Test data is for the final evaluation of our model. Our proposed method is composed of various phases. An overview of the proposed methodology is shown in Figure 1.

3.1. Image processing

First, we cropped the dark edges from the images and took only the brain portion from MRI images by using Open source Computer Vision (CV) Canny Edge Detection [15] technique. Canny Edge Detection is a multi-phase algorithm used to identify the edges of an object in an image. In Figure 2, The edges of the Real MRI brain have shown using the canny edge detection technique and then only the brain part of the image has been cropped.

3.2. Data augmentation

Data Augmentation is a strategy for artificially increasing the quantity and complexity of existing data [16]. We know that training a deep neural network needs a large amount of data to fine-tune the parameters. But our dataset is very small, so we applied the technique of data augmentation [17] on our training dataset by adding modifications to our images by making minor changes, such as flipping, rotation, and brightness. It will increase our training data size and our model will consider each of these small changes as a distinct image, and it will enable our model to learn better and perform well on unseen data. Figure 3 displays the numerous augmented images from a single image.

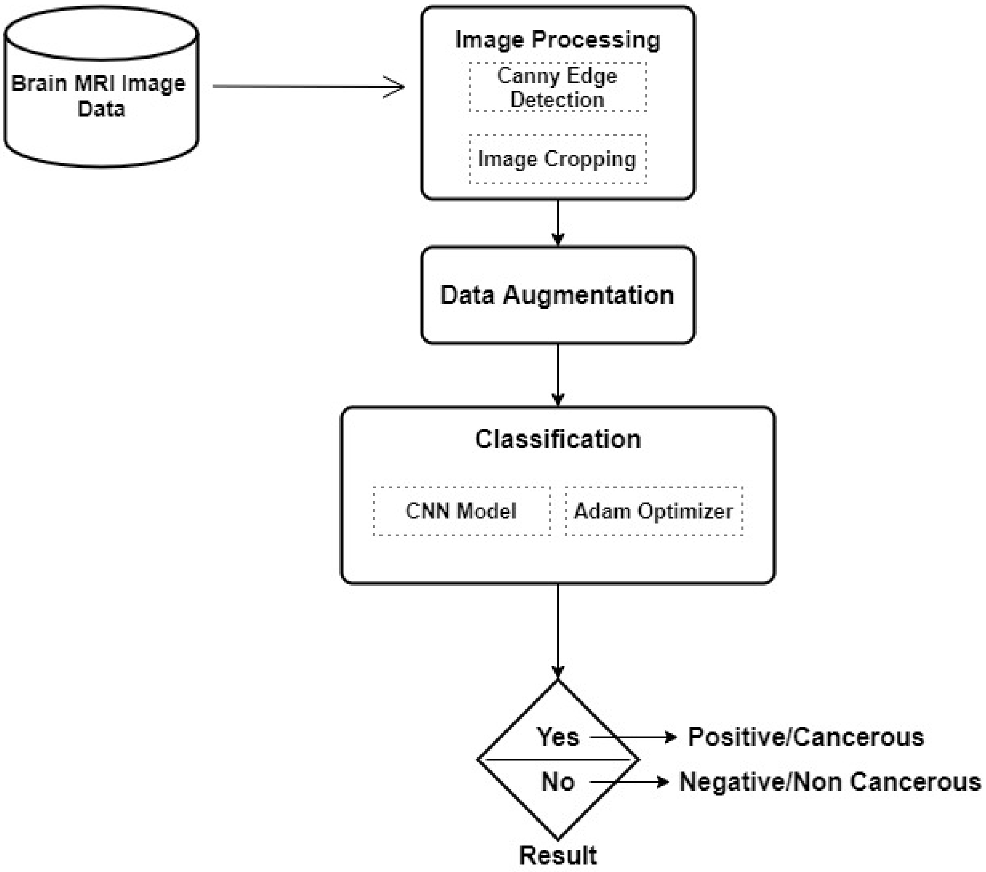

Figure 1. Overview of proposed methodology.

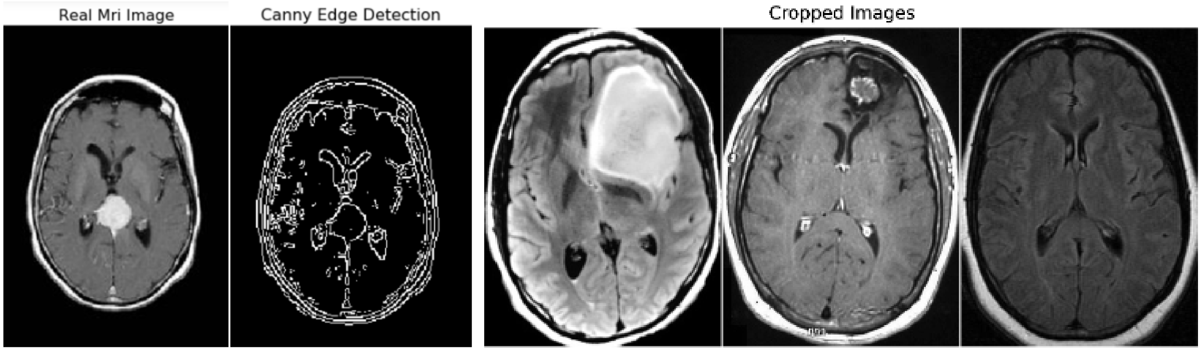

Figure 2. Finding edges using canny edge detection and cropping brain part.

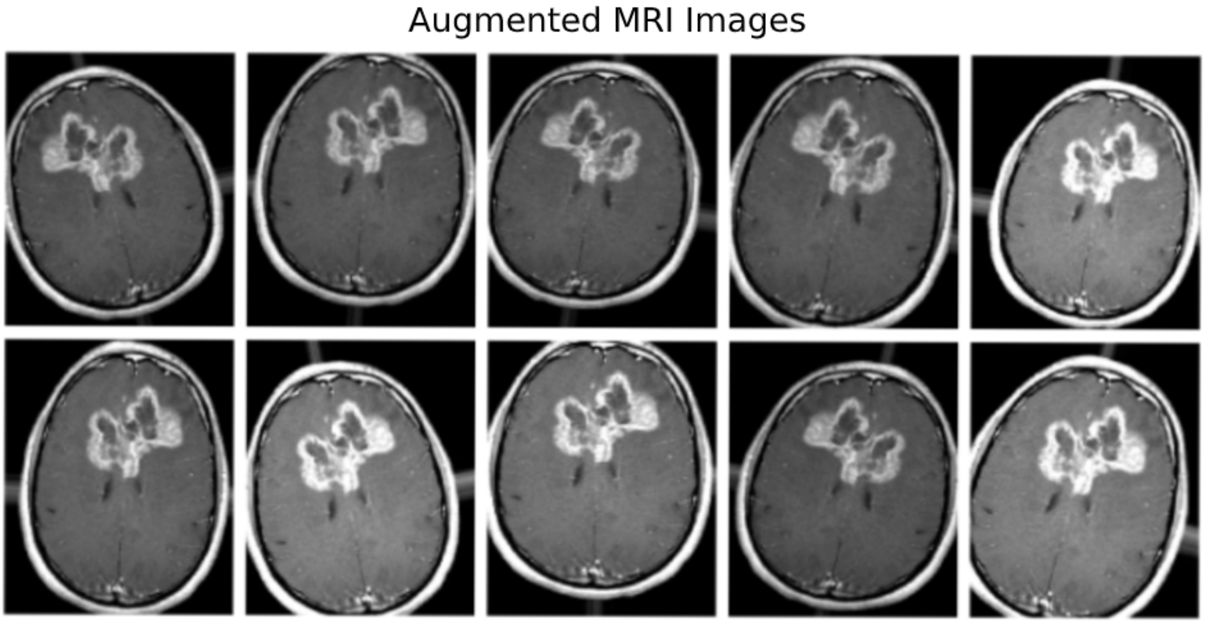

Figure 3. Applying data augmentation.

3.3. CNN Model

In our study, we proposed a simple CNN model, we extracted the augmented MRI image data of 224 × 224 input size having RGB Color channels with a batch size of 32 through our CNN model. Initially, we added a single 16 filters convolutional layer having a filter size of 3 × 3. The reason for placing a small number of filters as 16 is to detect edges, corners, and lines. And then a max-pooling layer with 2 × 2 filter was added on it to get the max summary of that image, then we increased the number of convolutional layers and the number of filters to 32, 64, and 128, having the same filter size of 3 × 3. This combines these small patterns as the number of filters increases and finds bigger patterns like a circle, a square, etc. And we applied max-pooling layers on top of those convolutional layers to get the most of it. Finally, we applied a fully connected dense layer of 256 neurons along with the softmax output layer that calculates the probability score for each class and classifies the final decision labels that either the input MRI image contains cancer or does not contain cancer in Yes or No. Figure 4 displays the layout of our proposed CNN architecture.

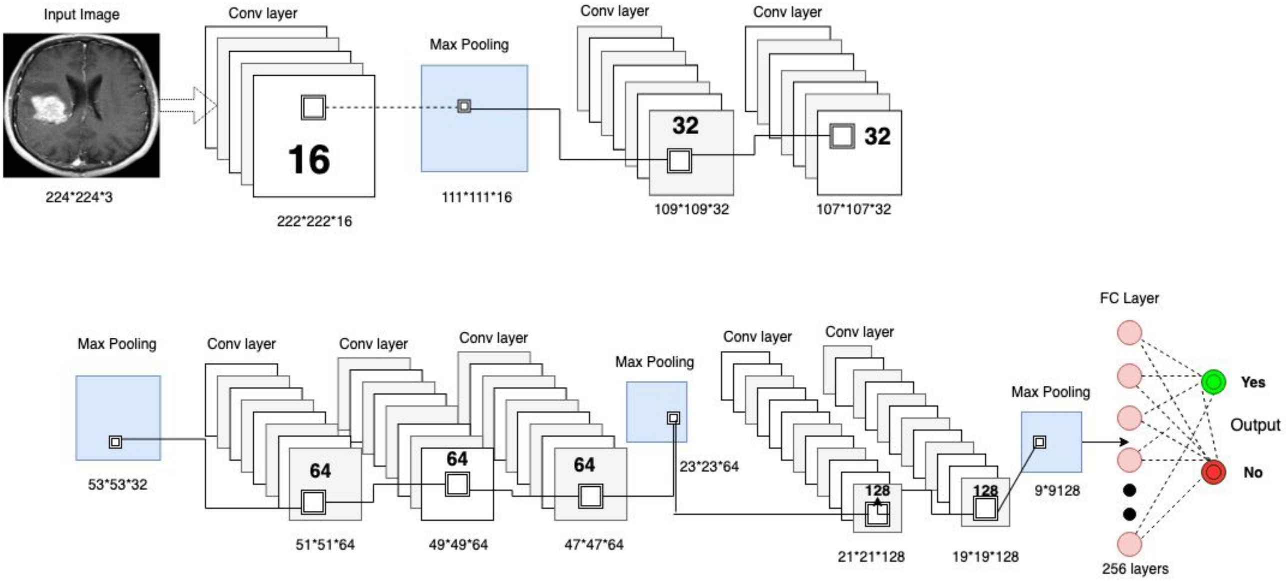

Figure 4. Proposed CNN model architecture.

We applied the Rectified Linear Unit (ReLU) activation function in each convolutional layer. An Activation function converts the input weighted sum into that node's output represented by Vinod and Hinton [18]. Rectifier Linear unit function is often used in hidden layers of the convolutional neural network. Mathematically, ReLU is represented by

$$f(z) = max(0, z) \tag{3.1}$$

Where z is the input when z is negative or equal to 0, it transforms the negative input to 0. When the input is greater than 0, then the output will be 1. So the derivative of ReLUs will be

$$f^{'}(z) = \begin{cases} 1, & for\ z \geq 0 \\ 0, & for\ z < 0 \end{cases} \tag{3.2}$$

So if the input is 0 then that neuron is a dead neuron in ReLU function and it won't be activated.

3.3.1. Loss function

In machine learning we use loss function to calculate the error between the algorithmic predicted values and the true label values. Then this error is minimized by using any optimization method. In our experiment we used Cross Entropy loss function presented by Shie Manor [19]. As we are doing binary classification of our MRI Images so we used binary cross entropy. In binary cross entropy error rate is calculated between 0 and 1. Mathematically it is represented as

$$J(z) = [y \log P(y) + (1 - y) \log(1 - P(y))] \tag{3.3}$$

Here y is the actual labels and P(y) is the predicted labels. So when the actual labels y will be 0 then the first term will be zero because y is multiplying with log. And when y will be 1 then second term will be zero (1-y) will be zero and it is multiplied with log. And if y = P(y) then J(y)(Loss) will be zero.

3.3.2. Optimization

In deep neural networks, we apply different optimization methods by adjusting parameters such as weights and learning rates to reduce the loss. In our experiment, we used Adaptive moment Estimation(Adam) optimizer, proposed by Diederik Kingma [20]. Adam optimizer is a combination of RMSprop and Stochastic Gradient Descent with momentum.

The Stochastic Gradient Descent method is proposed by Herbert and Sutton [21]. In simple Stochastic Gradient Descent, we take the derivative of weights, dW, and derivative of Bias, db for each epoch. And multiply with the learning rate.

$$\begin{aligned} W &= W - \eta \times dW \\ b &= b - \eta \times db \end{aligned} \tag{3.4}$$

While Stochastic Gradient Descent with momentum V is the moving mean of our gradients, here is the moving mean between 0 and 1 when we calculate dW and db on the current batch.

$$\begin{aligned} V_{dW} &= \beta \times V_{dW} + (1-\beta) \times dW \\ V_{db} &= \beta \times V_{db} + (1-\beta) \times db \end{aligned} \tag{3.5}$$

Similarly, Root Mean Squared Prop is an adaptive learning rate methodology presented by Geoff, Hinton [22]. In RMSProp, we take the exponential moving mean square of gradients. So in RMSProp,

$$\begin{aligned} S_{dW} &= \beta \times S_{dW} + (1-\beta) \times dW^2 \\ S_{db} &= \beta \times S_{db} + (1-\beta) \times db^2 \end{aligned} \tag{3.6}$$

Beta β, is a hyperparameter that controls exponentially weighted means. So combining the features of the weighted mean of past gradients and the weighted mean of the squares of the past gradients we implement Adam optimization technique, So the updated weights and bias in Adam optimizer will be

$$\begin{aligned} W &= W - \eta \times (V_{dw} / \sqrt{S_{dw} + \epsilon}) \\ b &= b - \eta \times (V_{db} / \sqrt{S_{db} + \epsilon}) \end{aligned} \tag{3.7}$$

Epsilon ϵ is a small number that prevents zero division (Epsilon $=10^{-8}$) and η is a learning rate with a different range of values.

3.4. Transfer learning

In deep learning, sometimes we use a transfer learning approach in which instead of making a scratched CNN model for the image classification problem, a pre-trained CNN model that is already modeled on a huge benchmark dataset like ImageNet is reused. Sinno Pan and Qiang Yang have introduced a framework for a better understanding of Transfer Learning [23]. Instead of starting the learning process from scratch, the transfer learning leverages previous learning.

3.4.1. VGG-16

In our experiment, we used a pre-trained VGG-16 convolutional neural network model which is fine-tuned by freezing some of the layers to avoid overfitting because our dataset is very small. VGG-16 is a CNN model of 16 Convolutional layers proposed in 2014 by Karen Simonyan and Andrew Zisserman [24]. The network image input shape is 224 × 224 × 3. It includes 16 Convolution layers with a fixed 3 × 3 filter size and 5 Max pooling layers of 2 × 2 size throughout the network. And at the top the 2 fully connected layers with a softmax output layer. VGG-16 Model is a large network, with approximately 138 million parameters. It's stacking many convolutional layers to build deep neural networks that improve the ability to learn hidden features. In Figure 5, the VGG-16 network architecture is shown.

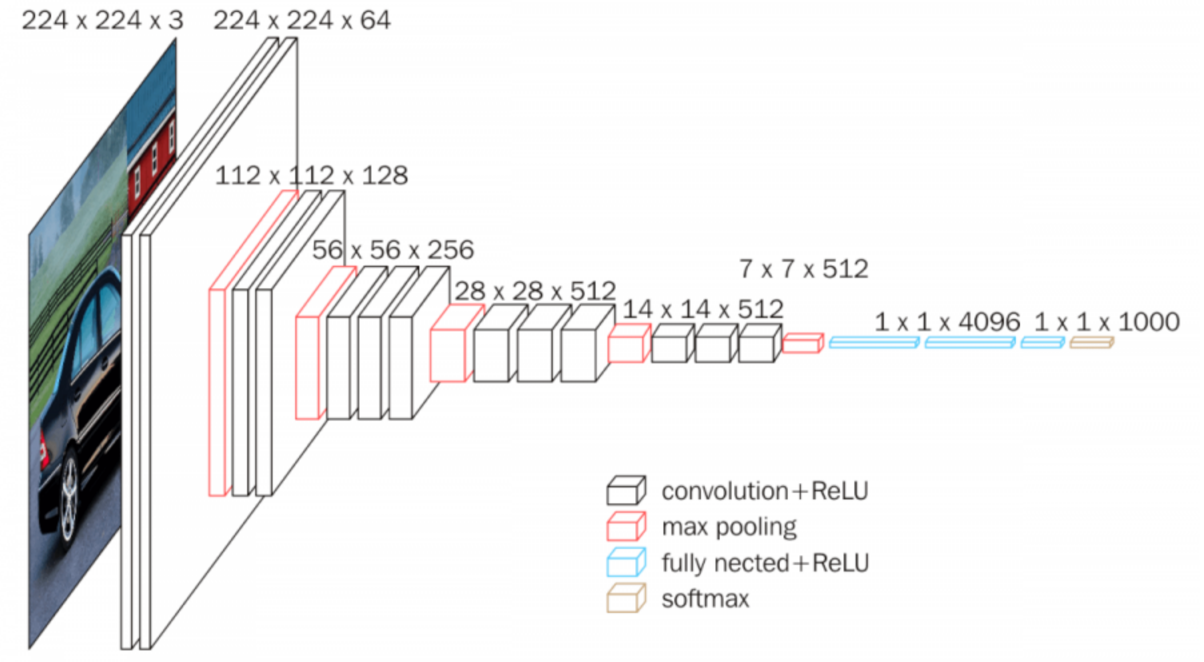

Figure 5. VGG-16 model architecture.

3.4.2. Inception-v3

The Inception network is also a pre-trained network model known as GoogleNet that was introduced by Google in 2014 [25]. Inception was a network of 22 layers having 5M parameters with a filter size of 1 × 1, 3 × 3 and 5 × 5 to extract features at various scales along with max pooling. The reason to use 1x1 filters is to save time for the computation. Google later on in 2015 scaled up Inception model to InceptionV3 [26], in which Convolutional layers are factorized to reduce parameters. Replacing the 5 × 5 Convolutional filters with two 3 × 3 filters to reduce computation without impacting the performance of networks. InceptionV3 model consists of 48 layers. In our experiment, we used the InceptionV3 model and fine-tuned the model according to the target data to avoid overfitting. In Figure 7, the architecture of InceptionV3 is shown.

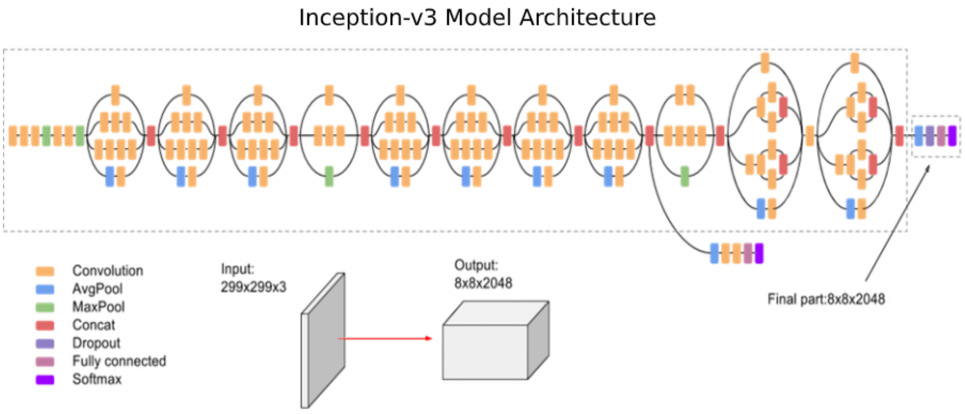

Figure 6. Inception-v3 model architecture.

3.4.3. ResNet50

ResNet50 is a 50-layer Residual Network with 26M parameters. The residual network is a deep convolutional neural network model that is introduced by Microsoft in 2015 [27]. In Residual network rather than learning features, we learn from residuals that are subtraction of features learned from the layer's inputs. ResNet used the skip connection to propagate information across layers. ResNet connects nth layer input directly to some (n+x)th layer which enables additional layers to be stacked and a to establish a deep network. We used a pre-trained ResNet50 model in our experiment and fine-tuned it. In Figure 8, the architecture of ResNet50 is shown.

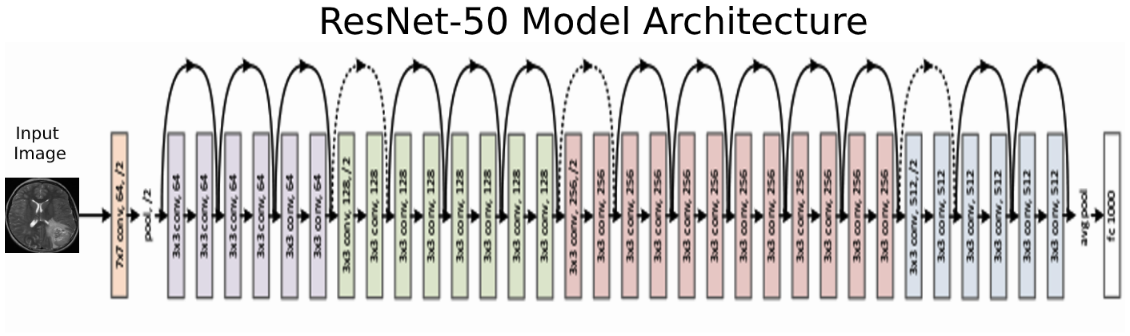

Figure 7. ResNet-50 model architecture.

4. Results and discussions

We experimented on brain tumor MRI Images dataset by Navoneel [14]. The dataset is publicly available, consists of 253 real brain images developed by radiologists using data from real affected patients. Its available on Kaggle, a shared data platform used for machine learning competitions. We split our data into training, validation, and testing. There are 185 images for training, 48 images for validation, 20 for testing to evaluate our model accuracy. First, data augmentation is done to enhance our dataset by doing minor changes in our MRI images and extract these augmented images from our proposed CNN model. We trained the models for 15 epochs with a batch size of 32. The experiment is done using TensorFlow and Keras libraries in python on a CPU having a 2.3 GHz core i5 processor with 8 Gb of ram. Our proposed model showed 96% accuracy on our training data and 89% accuracy on our Validation dataset. While using the transfer learning approach, we trained pre-trained VGG-16, ResNet-50, and Inception-v3 CNN models on the same dataset to compare the accuracy of our CNN model. VGG-16 showed 90% on training data and 87% accuracy on validation data, ResNet-50 showed 92% on training data and 87% on validation data and Inception-v3 showed 93% on training and 83% on validation data. Figure 8, displays the accuracy graph of the testing and validation phase during the iterations of our proposed CNN, VGG-16, ResNet-50, and Inception-v3 model.

We evaluated our model on unseen testing data. Here in Table 1, True Positive and True Negative categorize correct classification, where TP showing abnormal brain images positive and TN showing Normal brain images positive, while False Positive and False Negative categorize incorrect classification, while FP showing the normal brain images into a positive tumor and in FN showing the abnormal brain images into a negative tumor.

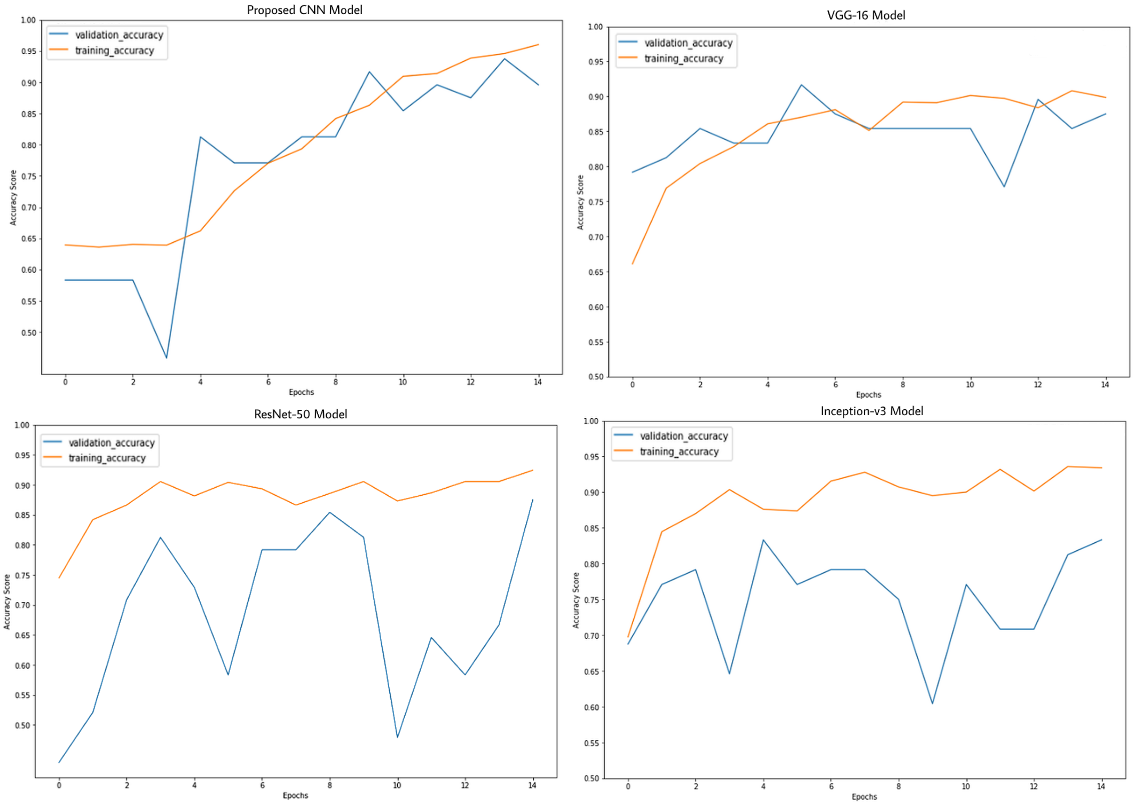

Figure 8. Accuracy graph of different models.

Table 1. Model classification performance.

Model	True Positive	True Negative	False Postive	False Negative
Proposed CNN	14	14	0	0
VGG-16	14	13	1	0
ResNet-50	13	12	2	1
Inception-v3	11	10	3	4

We calculated the Accuracy, Precision, Recall, and F1-Score of proposed CNN and other pre-trained models. Accuracy is the measurement of actual true classifications.

$$Accuracy = TruePositive(TP) + TrueNegative(TN)/(Total\ No\ of\ Samples) \tag{4.1}$$

Precision estimates how many positive labels we had predicted.

$$Precision = TruePositive(TP)/(TruePositive(TP) + FalsePositive(FP)) \tag{4.2}$$

Recall evaluates how many positive labels we had correctly predicted from our data.

$$Recall = TruePositive(TP)/(TruePositive(TP) + FalseNegative(FN)) \tag{4.3}$$

While F1-Score is the weighted mean of Recall and Precision.

$$F1 - Score = 2 \times (Precision \times Recall)/(Precision + Recall) \tag{4.4}$$

The Accuracy,Precision, Recall, and F1-score along with the training time of pre-trained models, and proposed CNN Model is shown in Table 2.

Table 2. Evaluation metric.

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC Score	Time(sec)
Proposed CNN	1.0	1.0	1.0	1.0	1.0	3085
VGG-16	0.96	0.93	1.0	0.97	0.96	6846
ResNet-50	0.89	0.87	0.93	0.90	0.89	9091
Inception-v3	0.75	0.77	0.71	0.74	0.75	5630

We plot the area under the ROC curve of all experimented models and compute the ROC-AUC score that is already shown in Table 2. ROC curve plot has False Positive Rate on X-axis and True Positive Rate on Y-axis. In Figure 9, the ROC Curve of experimented models is shown. True Positive Rate is known as Recall which tells the positive data labels which are predicted correctly as positive.

$$TruePositiveRate(TPR) = TruePositive(TP)/(TruePositive(TP) + FalseNegative(FP)) \tag{4.5}$$

While False Positive Rate tells the negative data labels which are predicted wrongly as positive.

$$FalsePositiveRate(FPR) = FalsePositive(FP)/(FalsePositive(FP) + TrueNegative(TN)) \tag{4.6}$$

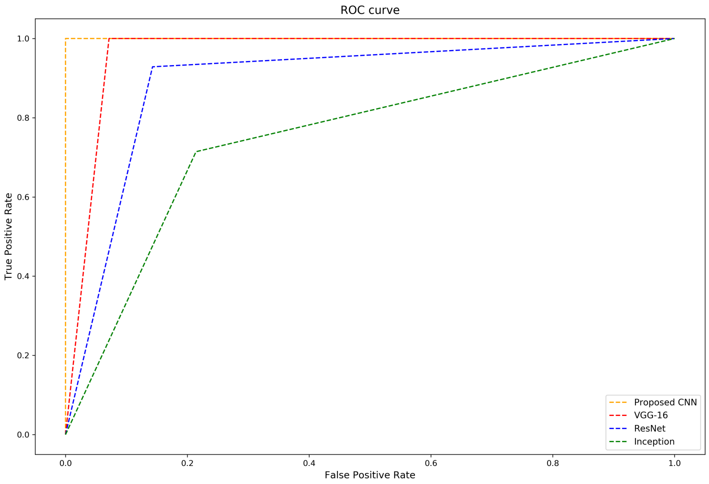

Figure 9. ROC-AUC curve graph.

5. Conclusions

In this paper, a new approach was presented to classify brain tumors. First, using the image edge detection technique, we find the region of interest in MRI images and cropped them then, we used the data augmentation technique for increasing the size of our training data. Second, we provide an efficient methodology for brain tumor classification by proposing a simple CNN network. For sophisticated and accurate results neural network requires a large amount of data to train on, but our experimental result shows that even on such a small dataset, we can attain full accuracy and our accuracy rate is very fine as compared to VGG-16, ResNet-50, and Inception-v3 model. Our model average training time per epoch is 205 sec while the VGG-16 takes 456 sec, ResNet-50 takes 606 sec and Inception-v3 takes 375 sec average training time per epoch. So, our model needs less computational specifications as it takes less execution time. Moreover, our model accuracy is much better than VGG-16, ResNet-50, and Inception-v3.

Our proposed system can play a prognostic significance in the detection of tumors in brain tumor patients. To further boost the model efficiency, comprehensive hyper-parameter tuning and a better preprocessing technique can be conceived. Our proposed system is for binary classification problems, however, in future work, the proposed method can be extended for categorical classification problems such as identification of brain tumor types such as Glioma, Meningioma, and Pituitary or may be used to detect other brain abnormalities. Also, our proposed system can play an effective role in the early diagnosis of dangerous disease in other clinical domains related to medical imaging, particularly lung cancer and breast cancer whose mortality rate is very high globally.We can prolong this approach in other scientific areas as well where there is a problem in the availability of large data or we can use the different transfer learning methods with the same proposed technique.

Acknowledgments

This work is supported by the project of manned space engineering technology(2018-14) Development of large-scale spacecraft flight and reentry surveillance and prediction system, and the National Natural Science Foundation of China (91530319), and the doctoral fund of the Southwest University of Science and Technology(13ZX7102).

Conflict of interests

The authors declare no conflict of interest.

References

1. NBTS, National Brain Tumor Society: Quick brain tumor facts, 2020. Available from: https://braintumor.org/brain-tumor-information/brain-tumor-facts/.
2. Cancer. Net, Brain Tumor: Staticstics, 2020. Available from: https://www.cancer.net/cancer-types/brain-tumor/statistics.
3. NHS, National Health Service: Brain Tumours, 2020. Available from: https://www.nhs.uk/conditions/brain-tumours/.
4. S. Basheera, M. S. S. Ram, Classification of brain tumors using deep features extracted using CNN, *J. Phys.*, **1172** (2019), 012016.
5. M. Sajjad, S. Khan, M. Khan, W. Wu, A. Ullah, S. W. Baik, Multi-grade brain tumor classification using deep CNN with extensive data augmentation, *J. Comput. Sci.*, **30** (2019), 174–182.
6. R. Carlo, C. Renato, C. Giuseppe, U. Lorenzo, I. Giovanni, S. Domenico, *Distinguishing Functional from Non-functional Pituitary Macroadenomas with a Machine Learning Analysis*, Mediterranean Conference on Medical and Biological Engineering and Computing, Springer, 2019, 1822–1829.
7. S. Khawaldeh, U. Pervaiz, A. Rafiq, R. Alkhawaldeh, Noninvasive grading of glioma tumor using magnetic resonance imaging with convolutional neural networks, *J. Appl. Sci.*, **8** (2018), 27.
8. N. Abiwinanda, M. Hanif, S. Hesaputra, A. Handayani, T. R. Mengko, *Brain tumor classification using convolutional neural network*, World Congress on Medical Physics and Biomedical Engineering, Springer, Singapore, 2019.
9. S. Das, R. Aranya, N. Labiba, *Brain tumor classification using convolutional neural network*, 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), 2019.
10. V. Romeo, R. Cuocolo, C. Ricciardi, L. Ugga, S. Cocozza, F. Verde, et al., Prediction of tumor grade and nodal status in oropharyngeal and oral cavity squamous-cell carcinoma using a radiomic approach, *Anticancer Res.*, **40** (2020), 271–280.
11. M. Talo, U. B. Baloglu, O. Yldrm, U. R. Acharya, Application of deep transfer learning for automated brain abnormality classification using MRI images, *Cognitive Systems Research*, **54** (2019), 176–188.

12. A. Rehman, S. Naz, M. I. Razzak, F. Akram, M. Imran, A Deep Learning-Based Framework for Automatic Brain Tumors Classification Using Transfer Learning, *Circuits Syst. Signal Process.*, **39** (2020), 757–775.

13. A. Cinar, M. Yldrm, Detection of tumors on brain MRI images using the hybrid convolutional neural network architecture, *Med. Hypotheses*, **139** (2020), 109684.

14. N. Chakrabarty, Brain MRI images dataset for brain tumor detection, Kaggle, 2019. Available from: https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection.

15. J. F. Canny, Canny Edge Detection, Open Source Computer Vision, OpenCV. Available from: https://docs.opencv.org/trunk/da/d22/tutorial_py_canny.html.

16. C. Shorten, T. M. Khoshgoftaa, A survey on image data augmentation for deep learning, *J. Big Data*, **6** (2019), 60.

17. Keras, Image data preprocessing keras API, Keras documentation. Available from: https://keras.io/api/preprocessing/image/.

18. V. Nair, G. Hinton, *Rectified linear units improve restricted boltzmann machines*, ICML, 2010.

19. S. Mannor, D. Peleg, R. Rubinstein, *The cross entropy method for classification*, Proceedings of the 22nd international conference on Machine learning, 2005.

20. D. P. Kingma, J. Ba, A method for stochastic optimization, arXiv: 1412.6980.

21. H. Robbins, S. Munro, A Stochastic Approximation Method, *Ann. Math. Stat.*, **22** (1951), 400–407.

22. G. Hinton, Neural networks for machine learning online course lecture 6a, Coursera. Available from: http://www.cs.toronto.edu/ tijmen/csc321/slides/lecture_slides_lec6.pdf.

23. S. J. Pan, Q. Yang, A survey on transfer learning, *IEEE Trans. Knowl. Data Eng.*, **22** (2010), 1345–1359.

24. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv: 1409.1556.

25. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, et al., *Going deeper with convolutions*, Proceedings of the IEEE conference on computer vision and pattern recognition, 2015.

26. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, *Rethinking the inception architecture for computer vision*, Proceedings of the IEEE conference on computer vision and pattern recognition, 2016.

27. K. He, X. Zhang, S. Ren, J. Sun, *Deep residual learning for image recognition*, Proceedings of the IEEE conference on computer vision and pattern recognition, 2016.