Heliyon 10 (2024) e36773

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

An efficient brain tumor detection and classification using pre-trained convolutional neural network models

K. Nishanth Rao [a], Osamah Ibrahim Khalaf [b], V. Krishnasree [c], Aruru Sai Kumar [c,*], Deema Mohammed Alsekait [d,*], S. Siva Priyanka [e], Ahmed Saleh Alattas [f], Diaa Salama AbdElminaam [g,h]

[a] Department of ECE, MLR Institute of Technology, Hyderabad, India
[b] Department of Solar, Al-Nahrain Research Center for Renewable Energy, Al-Nahrain University, Jadriya, Baghdad, Iraq
[c] Department of ECE, VNR Vignana Jyothi Institute of Engineering and Technology, Telangana, India
[d] Department of Computer Science and Information Technology, Applied College, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia
[e] Department of ECE, Chaitanya Bharathi Institute of Technology, Telangana, India
[f] Information Science Department, Faculty of Arts & Humanities, King Abdul-Aziz University, Jeddah, Saudi Arabia
[g] MEU Research Unit, Middle East University, Amman 11831, Jordan
[h] Jadara Research Center, Jadara University, Irbid, 21110, Jordan

ARTICLE INFO

Keywords:
Brain tumor detection
MRI scan
Convolution neural networks (CNN)
Deep learning and data augmentation

ABSTRACT

In cases of brain tumors, some brain cells experience abnormal and rapid growth, leading to the development of tumors. Brain tumors represent a significant source of illness affecting the brain. Magnetic Resonance Imaging (MRI) stands as a well-established and coherent diagnostic method for brain cancer detection. However, the resulting MRI scans produce a vast number of images, which require thorough examination by radiologists. Manual assessment of these images consumes considerable time and may result in inaccuracies in cancer detection. Recently, deep learning has emerged as a reliable tool for decision-making tasks across various domains, including finance, medicine, cybersecurity, agriculture, and forensics. In the context of brain cancer diagnosis, Deep Learning and Machine Learning algorithms applied to MRI data enable rapid prognosis. However, achieving higher accuracy is crucial for providing appropriate treatment to patients and facilitating prompt decision-making by radiologists. To address this, we propose the use of Convolutional Neural Networks (CNN) for brain tumor detection. Our approach utilizes a dataset consisting of two classes: three representing different tumor types and one representing non-tumor samples. We present a model that leverages pre-trained CNNs to categorize brain cancer cases. Additionally, data augmentation techniques are employed to augment the dataset size. The effectiveness of our proposed CNN model is evaluated through various metrics, including validation loss, confusion matrix, and overall loss. The proposed approach employing ResNet50 and EfficientNet demonstrated higher levels of accuracy, precision, and recall in detecting brain tumors.

* Corresponding authors.
E-mail addresses: asaikumar.nitw@gmail.com (A.S. Kumar), Dmalsekait@pnu.edu.sa (D.M. Alsekait).

https://doi.org/10.1016/j.heliyon.2024.e36773
Received 27 December 2023; Received in revised form 21 August 2024; Accepted 22 August 2024

Available online 26 August 2024

1. Introduction

The brain is the most crucial organ since it regulates the activity of every other organ and has a role in decision-making [1]. This primary hub of the central nervous system regulates all bodily processes, both voluntary and involuntary, that take place every day. An uncontrolled proliferation of fibrous networks of abnormal brain tissue is what we call a tumor. Approximately 3,540 youngsters will receive a brain tumor diagnosis this year at the age of fifteen. Recognizing the signs of a brain tumor and its progression is crucial for both prevention and treatment. Radiologists frequently employ magnetic resonance imaging (MRI) for this purpose while examining brain malignancies. Applying deep learning techniques, the study conducted in this paper determines if the brain is healthy or ill [2].

1.1. Detection of brain tumor

The body's most pivotal and eloquent organ is the brain. One of the prominent reasons for illness of the brain is tumors. Rampant growth of cells in the brain is called a tumor. One of the most risky and deadly diseases affecting the mankind is the brain tumor. Nearly 11,700 people are diagnosed with a brain tumor every year. The distinguishing of brain tumors is carried out by magnetic resonance imaging (MRI) effectively. Enormous data will be generated during the scanning process which is ready for the examination by the radiologists. The diagnosis process done by human beings is time taking and may lead to erroneous outcomes sometimes [3].

In modern years, the edge cutting technologies like machine learning and deep learning is adopted to make intelligent and accurate decisions in critical fields like finance, medical etc. These algorithms have proved to have better decision makings, speedy results as well as avoiding manual intervention. And the consistency and reliability also have been improved tremendously. An automatic classifier was developed for detecting and classifying the brain cancers from images taken by MRI scan. Number of Artificial based algorithms are available in literature to do the task of auto detection and classification. We supposed a method of deep learning called for the finding of tumors from the MRI scans [4].

1.2. Convolutional neural network

By using artificial neural networks with more than one internal layer along with learning mechanism deep learning algorithms have been formed. The main merits of these algorithms are handling capacity of huge volumes of data, less time taken for decision making and greater accuracies. That's why these algorithms have acquired huge popularity. Usage of more number of internal layers overcome the disadvantages of earlier techniques in various fields, most predominantly image processing. Convolutional Neural Networks (CNNs) are widely used deep neural networks regularly operated for image processing. Detection of objects from images, segmentation of images and classifying them are the tasks performed by CNNs [5]. A CNN comprises of a convolutional layer that performs the feature recognition and detection and the outcomes of the convolutional layers are used to forecast the final decision or type of class of the image. In machine learning issues, the CNN is very opt for applications associated with images [6].

1.3. Transfer learning

The use of weights from a pre-trained on a dataset is referred to as "transfer learning" [7]. In this work, 1000 classes were defined using weights from the model that was trained using the ImageNet Dataset. Initially for a given task, a model is trained with designated data which is large in size. This model is used to train the data for another task where data size is small. This is the core principle. We use patterns that were established by finishing a similar task rather than starting the learning process from scratch. Transfer learning has several benefits, but the three that matter most are that it takes less time to train, performs better on neural networks, and doesn't need a lot of data. Transfer learning is useful in situations where access to vast amounts of data is not always attainable but is nevertheless necessary to train a neural network from the beginning. Transfer learning enables pre-training of the model, which enables the development of a workable machine learning model with less training data. A deep neural network may require days to train from scratch on a difficult problem. As a result, training time is also reduced. It has been applied to employ ResNet50 and EfficientNet pre-trained models.

1.4. Augmentation of data

To train the model of machine learning one of the crucial and critical components is the dataset. But practically obtaining a huge dataset of brain tumor is not possible. Hence if it is required large number to train the model, one needs to carry out the augmentation. By altering the data the available dataset is expanded artificially is called the data augmentation. By rotating, shearing, flipping, shifting and cropping the images of dataset can be re-oriented and recreated as augmented data set. Due to big number of images are considered for training the accuracy of the model can be aggravated enormously.

1.5. Programming languages and tools used

1.5.1. Jupyter notebook

Jupyter Notebook, has become a widely used open-source interactive web application for creating and sharing documents that contain live code, equations, visualizations, and narrative text. Developed in Python, it supports various programming languages through its interactive computing framework. Jupyter Notebooks are organized into cells, allowing users to execute code in a step-by-step fashion, enhancing the clarity of code development and analysis. It has gained popularity in academia, data science, machine learning, and scientific research due to its versatility and integration with libraries like NumPy, Pandas, and Matplotlib. For creating and sharing the documents "Jupyter Notebook" can be used. Equations can be created, text can be narrated, and code can be written and visualized. Users are permitted to compose the constituents of a data project using the Jupyter notebook in one place, made easy the process for explanation.

1.5.2. Python

Convolutional Neural Networks (CNNs) have played a pivotal role in revolutionizing the field of computer vision, and Python has been a prominent language for implementing and developing CNNs. Object-oriented programmes can be implicitly generated by Python. No need of selecting the method, specifying a function, declaring parameter or type in the source code. Due to this the length of the code is small and versatile. One of the main merit of the Python is it gives in a variety of libraries and frameworks. Python's extensive libraries, such as TensorFlow and PyTorch, have become go-to tools for building and training CNN models. TensorFlow, an open-source machine learning library, offers a high-level API, Keras, simplifying the creation of CNN architectures. Researchers and developers leverage TensorFlow to design intricate neural networks for image classification, object detection, and other vision tasks. PyTorch, another popular deep learning framework, has gained traction for its dynamic computation graph and intuitive interface. It facilitates the construction of CNNs with flexibility, making it a preferred choice for both academia and industry.

Data augmentation is preferred over relying on a large database in brain tumor detection due to several key reasons. First, acquiring a large, well-annotated MRI image database is often challenging, costly, and time-consuming, given the sensitive nature of medical data and the specialized requirements for accurate labeling. Data augmentation, on the other hand, allows for the artificial expansion of the existing dataset by applying transformations such as rotation, flipping, and scaling, which enhances the diversity of the training data. This not only helps improve the model's ability to generalize to new, unseen data but also reduces the risk of overfitting, where the model might perform well on training data but poorly on new data. Additionally, data augmentation is more computationally efficient, enabling the development of robust deep learning models without the need for an extensive database, which would otherwise require significant resources to manage.

The existing approach for brain tumor detection relies on manual examination of MRI scans by radiologists, a process that is both time-consuming and susceptible to inaccuracies due to human error and fatigue. In contrast, the proposed approach leverages Convolutional Neural Networks (CNNs) to automate the analysis of MRI images, significantly enhancing efficiency and accuracy. By employing pre-trained CNNs and utilizing data augmentation techniques, the proposed method reduces the time required for diagnosis and improves the precision of tumor detection. Metrics such as validation loss, confusion matrix, accuracy, precision, and recall are used to evaluate the effectiveness of this automated approach, offering a more objective and reliable means of detecting brain tumors compared to traditional manual examination.

The paper presents efficient brain tumor detection and classification using pre-trained CNN models. The major novel contributions of the proposed work are as follows:

- To start with brain tumor classification, relevant libraries are imported for machine learning algorithms.
- The data is splitted into training, validation and test sets. Accordingly, augmentation is necessary if an enormous amount of data is needed to train the model. An artificial expansion of the given dataset is achieved through data augmentation, which entails modifying the data.
- In this proposed brain tumor detection, precise segmentation is crucial for identifying the extent and boundaries of tumors, aiding in accurate diagnosis and treatment planning using GrabCut algorithm to accurately segment objects in images.
- In the proposed CNN approach for image classification, specifically in the context of brain tumor classification, both ResNet and EfficientNet models were utilized. The proposed CNN model was then applied to train the data across multiple epochs.
- The effectiveness of the proposed CNN model was assessed through the generation of diverse plots, including those depicting validation loss, confusion matrix, and overall loss. When compared to related works, both with and without data augmentation, the proposed method demonstrated higher accuracy in detecting tumors.
- Higher levels of accuracy, precision, and recall were demonstrated by the proposed approach using ResNet50 and EfficientNet to identify brain tumors.

The organization of this research is determined as follows; Section 2 provides the related work and, Section 3 illustrated the proposed methodology of detecting brain tumors. The experimental outcomes are represented in Section 4. The research paper concludes with Section 5.

2. Related work

By combining a self-defined architecture of artificial neural networks (ANNs) and convolutional neural networks (CNNs), Gokila brinda et al. [8] suggested a method for detecting brain tumors in magnetic resonance imaging (MRI) scans. This instance made use of a seven-layer ANN model. In order to train and validate the dataset, two hundred epochs are utilized. The 65.21% accuracy rate was the outcome of the test. To improve the accuracy a massive amount of images. One of the top techniques for extracting image datasets is CNN. By decreasing the image size while preserving the information required for prediction, the CNN is able to foresee future events. M. Aarthilakshmi et al., [9] suggested brain tumor detection using machine learning. Here images were pre processed by MATLAB. These categorical variables are given to 4 classification algorithms using WEKA 3.6. It computes precision/recall, the F-measure, the percentage of appropriately classified images and the time consumed to produce each model. By doing pre-processing on the MRI scans, the accuracy of the classification has been increased using Neural networks. Dheiver et al., [10] presumed brain tumor detection using deep learning-based, Depth-wise separable CNN. The Keras library is doing the required pre processing using “Image Data Generator”. Depth-wise separable convolutions are employed in “Mobile Net architecture”. Training of model is carried out for one fifty epochs to learn different complex patterns and obtained 92% accuracy. They reported that Depth-wise Separable Convolution Neural Network was accurate than K-NN, Support Vector Machine and CNN.

Javaria Amin et al., [11] proposed the automatic brain tumor detection is very critical since precise accuracy is essential when human lives are at stake. Malignancies are automatically detected using feature extraction and classification in a machine learning algorithm from MRI scans. This study suggests that how MRI images are preprocessed, segmented, and automatically identified the tumors. “Principal Component Analysis” and “Probabilistic Neural Network (PNN)” were claimed for the classification by Sonali B. Gaikwad et al., [12]. The presumptive method used the PNN classifier to automatically detect cancers and categorize them as benign or normal. The malignant picture can be used to further identify the meningioma and glioma tumor types with an accuracy of 97.14%. In terms of accuracy, retraining time, training period, and roughness change in weights, the PNN classifier was victorious. When dealing with classification problems, PNN is frequently used. The simplicity and speed of PNN are the main reasons for its popularity. Using a CNN and NADE hybrid model, Raheleh Hashemzehi et al., [13] presented deep learning for the first time for the purpose of detecting brain tumors in MRI scans. The three steps of learning for hybrid architecture are feature exploitation, categorization, and density estimation. A distribution estimator, two convolutional neural networks (CNNs), and a fully linked network make up the four components of the model. In order to choose the right joint distribution during density estimation, the NADE model is trained. A classification model for magnetic resonance imaging (MRI) images of brain tumors was developed in this study. It makes use of a hybrid structure and convolutional neural network (CNN) architecture to exclude unwanted features and to smooth out the tumor border

For precise lesion symptom segmentation, Tanzila Saba et al., [14] proposed the grab cut method which is a fine-tuned transfer learning model called VGG-19 is used to acquire features. These features are then serially combined with hand-crafted features like shape and texture. Classifiers are given a fused vector once these features have been adjusted using entropy for rapid and accurate classification. These are the main databases used to test the approach on top of medical image computing and computer-assisted intervention (MICCAI): Multimodal brain tumor segmentation (BRATS) 2015, 2016, and 2017. To improve accuracy and create appropriate classification decisions, Sharan Kumar et al., [15] present an optimized deep learning method termed Dolphin-SCA based Deep CNN. A pre-processing step is performed on the input MRI images before they are passed on to the segmentation method. Fuzzy deformable fusion models using Dolphin Echolocation based Sine Cosine Algorithms (Dolphin-SCA) are used to carry out the segmentation process. The next step is to extract features using power LDP and statistical characteristics such as skewness, variance, and mean. In order to classify brain tumors, the features that were retrieved are fed into a Deep Convolution Neural Network (Deep CNN) that was trained using Dolphin-SCA.

3. Proposed methodology

3.1. Dataset

Navoneel Chakrabarty’s dataset has 253 Brain MRI scans. The images are divided into two files, yes and no, each containing images with and without brain tumors. This dataset is used for the detection of brain tumor. The Kaggle dataset for brain tumor imaging was used. This dataset is a combination of datasets from “figshare”, “SARTAJ”, and “Br35H”. It has 7022 MRI scans of the human brain divided into four categories: glioma, meningioma, no tumor, and pituitary as depicted in Fig. 1. This dataset is used for classification of tumors [16,17].

3.2. One hot encoding

The processing and feeding of diverse varieties of inputs model along with the nature of model, is deciding the rendition of machine learning. The maximum number of machine learning algorithms takes only numerical values as inputs; because of this reason, features must be converted into numerical values. Now the algorithms interpret and excerpt useful data from these features. Some of the encoding techniques are simple encoding, target based encoding, dummy-based encoding and hot dummy encoding. Two binary variables are assigned with two scenarios as “0” for the no category available and “1” for availability of category.

3.3. Model building for detection of tumor

The sequential model and the functional model are the two of numerous models used by CNN. Both the Sequential model and the Functional model are equally proficient in their respective areas. It supports branching and layer sharing in addition to accepting numerous inputs and outputs. In sequential model every layer fed with one input and generates one output and all layers are piled to create the complete network. For most of the matters of course, the sequential application programming interface permits us to produce models layer by layer. The main constraint of this is that it averts us from generating models with many inputs / outputs or shared layers [18–20].

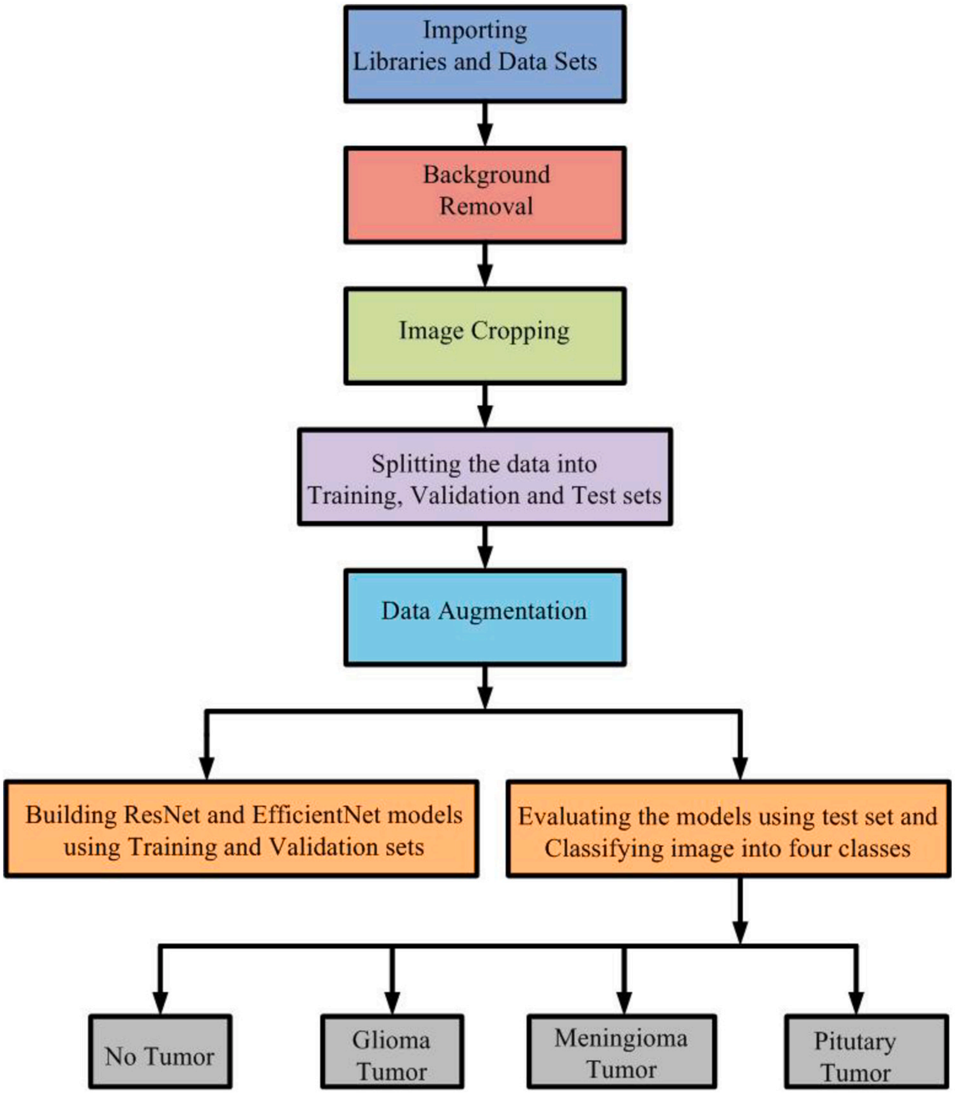

Fig. 1. Block diagram of Brain Tumor Classification.

The features of input image are extracted by convolution layers. The input image is convolved with the suitable kernel. The size of the input image determines the size of the kernel. The kernel used for convolving masks the input image and it is in matrix form hence it is called as convolution matrix or a convolution mask. The main aim of using a pooling layer is to reduce the dimensions of the input image and to achieve the dimensionality reduction in the network. Pooling minimizes the size of the sample by removing the redundant information and transfers the needed to the successive layers of CNN [21–23]. The block diagram of proposed method using CNN is depicted as shown in Fig. 2.

The Fully Connected Layers are an integral part of feed-forward neural networks. Before being utilized as an input, the final output of the pooling or convolutional layer is passed on to the fully connected layer, where it is rounded. If you want to prevent "over fitting" on your training data, you can use the dropout layer to discourage neurons from helping out later layers. Another technique that is frequently utilized in deep learning is batch normalization. Both the input and the output of intermediate stages undergo this normalizing process [24–26]. In a convolutional neural network (CNN), the output layer is a fully linked layer that receives input from the previous layers and processes it in a way that converts it into the required number of classes. The "Loss function" is applied using "categorical cross entropy," and the "adamax optimizer" is used as the optimizer.

In order to achieve better accuracy, the fitting technique involves altering the model's features. Data is processed by use of an algorithm. Accuracy is measured by correlating the model's output with the actual values of the dependent variable. In machine learning, an epoch is the total number of iterations of the model that use the whole training dataset. Thirty, forty, and fifty epochs were considered for the model's training. The amount of samples that are sent via the network is determined by the batch size [27–29]. A batch size of twenty five was used in this work. To observe the output of any Neural Network during the process of training, the best

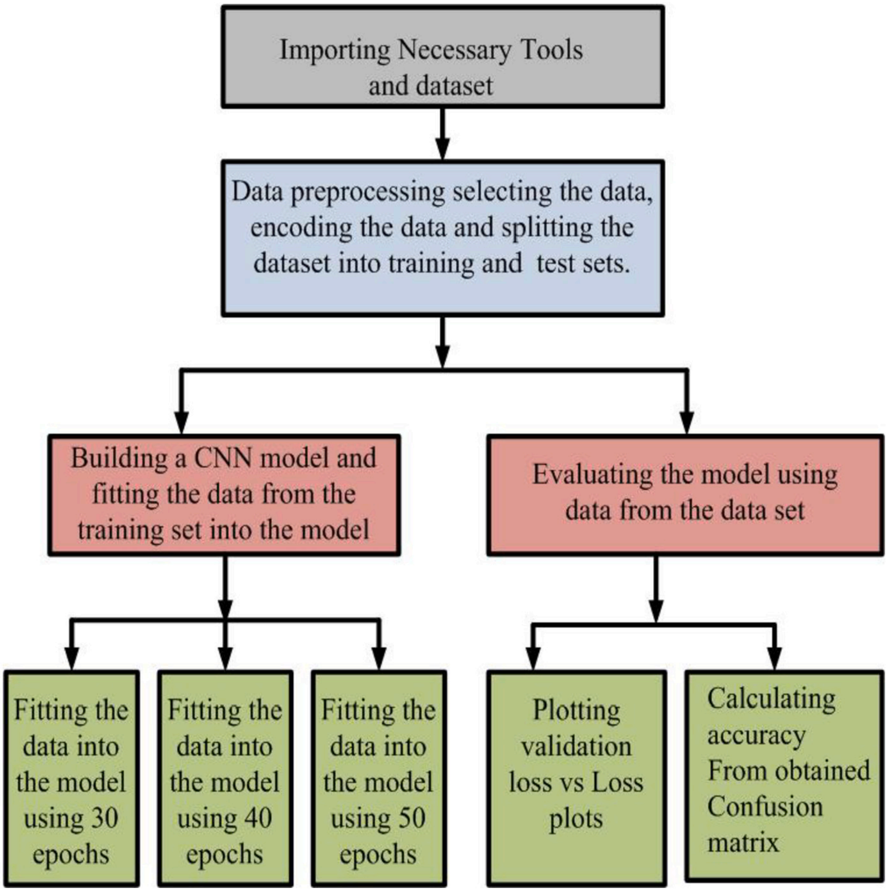

Fig. 2. Block Diagram of the proposed model using CNN.

option is verbose. The fitting of Model was carried out in two stages, one before augmenting the data and another after augmenting the data.

3.4. Grabcut algorithm

The Grabcut algorithm is used to separate the foreground image from the background image. The foreground image is the subject that we are interested in and we train our model on this foreground image. This reduces unnecessary background noise which helps our model train well. In this algorithm, the model draws rectangles which are referred to as Region of Interest (ROI). ROI given by the user encloses the subject we are looking for. Mainly, the segmentation amount between the foreground and the background will be decided by the user, thereby deciding our ROI. Then a Gaussian Mixture Model (GMM) is performed on this ROI which segregates the ROI pixels into two categories, the Source node, and the Sink node. Pixels related to the foreground image belong to the Source node and the rest of the pixels will be assigned to the Sink node. The pixels along the edge of the foreground and background are assigned weights according to the probability that the pixel belongs to the foreground or background. These assigned weights are summated to give us the cost function and an algorithm is performed to segment the graph of the Source and Sink node. After segmentation, Source node pixels are labeled as foreground and Sink node pixels are labeled as background.

3.5. Model building for classification of tumors

As a pre-processing step the cropping of images was carried out and the removal of pixels surrounding the image was carried out. Then the data base size expansion was done on the data set to increase the size. The extraction of features was carried out from MRI images with the help of pre-trained convolutional layers [30–32]. Later, the extracted features are applied to the newly developed, fully connected artificial neural network for classifying the input MRI brain images. The next models we created were trained and used for classifying the actual data of brain MRI images. Our study made use of the ResNet50 and EfficientNet models.

3.5.1. ResNet50

The residual network was first proposed in 2015 by Microsoft Research specialists. The residual block concept was developed by this design to solve the gradient vanishing or exploding issue. In this network, it employs a technique called skip connections. The skip connection connects layer activations to the succeeding layers by skipping a few levels in between. A residual block is created as a result of this. These leftover pieces are stacked to create ResNets.

The Vanishing Gradient Problem in Machine Learning appears during the back propagation training of neural networks. The weights of a neural network are updated using the back propagation process. To lessen the model's loss, each weight is altered using the back propagation method. This issue makes it challenging to learn and adjust the parameters of the network's earlier layers. The vanishing gradients problem is one form of unstable behavior we will encounter while training a deep neural network. It depicts the situation in which a deep multilayer feed-forward network fails to convey useful gradient data from the model's output layers to its input end. Regularization will skip any layers that have a negative impact on the architecture's performance by creating this type of skip connection. This avoids the issues that can arise when training a highly deep neural network with vanishing or exploding gradients.

The original ResNet design, ResNet-34, consisted of 34 weighted layers. The architecture included nine layers with 33 and 64 kernels, a max pooling layer with a two-sized stride, and a convolution layer with 64 kernels. Additionally, there were three sets of repeated sequences: the first included layers with 11,64 kernels, the second had layers with 11,256 kernels, and the third comprised layers with 11, 256 kernels. In the 50-layer ResNet, this sequence was iterated three times. The model also included 12 layers with 11, 128 kernels, another layer with 33, 128 kernels, and a third with 11, 512 kernels, repeated four times. Moreover, there were 18 layers with 11, 256 kernels, another with 33, 256 kernels, and 11, 1024 kernels, repeated six times. The final part of the architecture involved nine layers, each consisting of 11, 512 kernels, 33, 512 kernels, and 11, 2048 kernels, repeated three times. This intricate structure contributed to the network's ability to capture complex features and representations, making it effective for various computer vision tasks. Both ResNet34 and ResNet50 have significantly contributed to the success of deep learning in computer vision tasks. The introduction of residual connections allows for the training of very deep networks, mitigating issues like vanishing gradients and enabling the development of highly accurate models for image classification and other vision-related tasks.

3.5.2. EfficientNet

With the growing need for convolution neural networks in the field of image classification, one of the main concerns is the efficiency of the given network. To improve the model efficiency, the depth of the network is deepened. This is known as depth scaling. This works up to a certain extent, but as we go on increasing the depth of the network, there is saturation that is achieved, i.e., the accuracy of the model remains constant even though we increase the depth. This is due to the vanishing gradient problem.

To overcome this problem up to a certain extent, EfficientNet is developed. In EfficientNet, our focus is not only on depth scaling but also on other factors like Resolution scaling and Width scaling. In Resolution scaling, we take images of higher resolution to work on complex features present in the image. This complex feature extraction helps our model to perform better as we can draw more information compared to low-resolution images. In Width scaling, to extract more features from the images, we need to have more feature maps or a greater number of channels. As we are processing higher-resolution images, we need to work with denser networks. Hence depth scaling also plays a major role here. It has been observed that increasing resolution, depth, and width increases our model accuracy, but for bigger models, the gain diminishes. To work with our model more efficiently, it is important to balance the network width, depth, and resolution. The factor up to which we can increase the resolution, width, and depth of the network is decided by compound scaling. In compound scaling, we take a baseline model to generate other efficient models. In EfficientNet, the baseline model that we choose is the Efficient-B0 network. This was developed by Neural Architecture Search (NAS). The depth, width, and resolution factors are considered as α, β, and γ, respectively. By performing a grid search, we get constant values of α, β, and γ as 1.20, 1.10, and 1.15 respectively. It explains that if we increase the resolution of our model by 15%, we need to perform width scaling by 10% and depth scaling by 20%.

3.5.3. Fully connected layers

We use fully connected layers for classification. In our model, we need to classify only four classes. Hence the pre-trained model architectures utilized cannot be used because they have 1000 output classes. The new architecture was defined, and it consists of a global average pooling layer, two dropout layers, and two dense layers with relu and softmax activation functions.

3.6. Model evaluation

To know about the workings of a machine learning model, it has to be evaluated so that its merits and demerits can be understood. The fit of training data set is reflected as the training loss. Some part of the dataset reserved to verify the performance of the model is termed the validation set. The validation loss indicates and evaluates a deep learning model's performance on the validation data set. A confusion matrix is a table to decide how best a classification system works. Numerous parameters of the model, like True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) and accuracy are computed using confusion matrix.

4. Result analysis

The proposed methodology is implemented using Jupyter notebook. The section outlines the detection of brain tumor using a CNN model by inserting the data obtained from the trained data set.

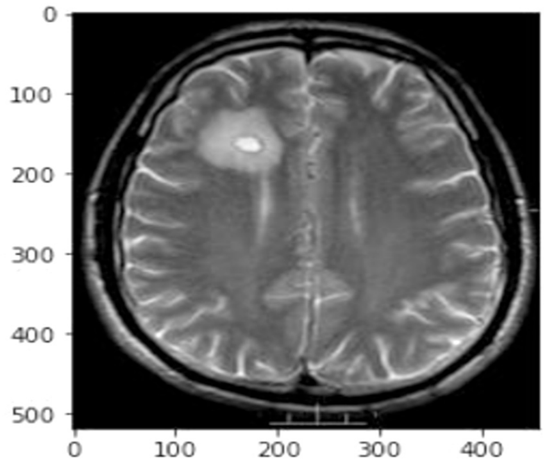

Fig. 3. Result of Output when the image with tumor is given as input.

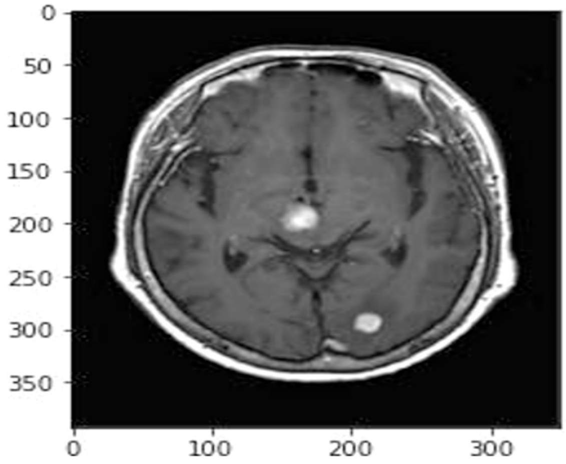

Fig. 4. Result of Output of the model when image without tumor is given as input.

4.1. *Detection of tumor without data augmentation*

Detecting tumors in medical images without data augmentation is certainly possible, but data augmentation is a common technique used to artificially increase the size of the training dataset, which can be especially beneficial when the available dataset is limited.

The results of output when the image with tumor and without tumor are given as inputs is depicted in Fig. 3 and Fig. 4.

4.1.1. *Training loss vs validation loss plot*

For epoch sizes of thirty, forty and fifty, validation loss and loss plots are plotted as depicted in Fig. 5, Fig. 6 and Fig. 7. For fifty epochs, training loss is nearly zero but validation loss is very high. This shows that the model is not properly fitted. Under fitting happens whenever the model is unable to model the training set properly, resulting in high errors.

4.1.2. *Validation accuracy vs training accuracy plot*

Training accuracy and validation accuracies are plotted observed for epoch size of thirty, forty and fifty as depicted in Fig. 8, Fig. 9 and Fig. 10.

4.2. *Detection of tumor with data augmentation*

Using data augmentation can be a valuable strategy for improving the performance of a tumor detection model, especially when the available dataset is limited. Data augmentation involves applying various transformations to the original images to artificially increase the diversity of the training dataset. This can help the model generalize better to different variations of the input data. Data augmentation, when used appropriately, can enhance the robustness and generalization of the model, leading to improved tumor detection performance. Experiment with different augmentation techniques to find the combination that works best for your specific

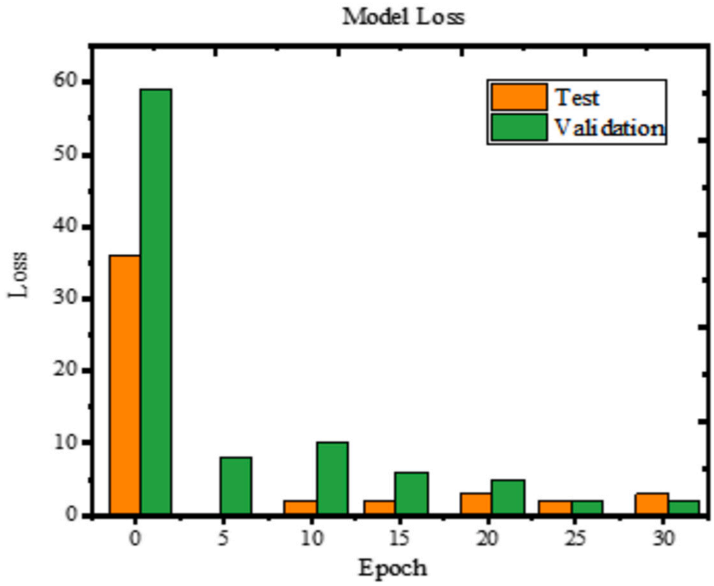

Fig. 5. Validation loss Vs loss plot for 30 epochs.

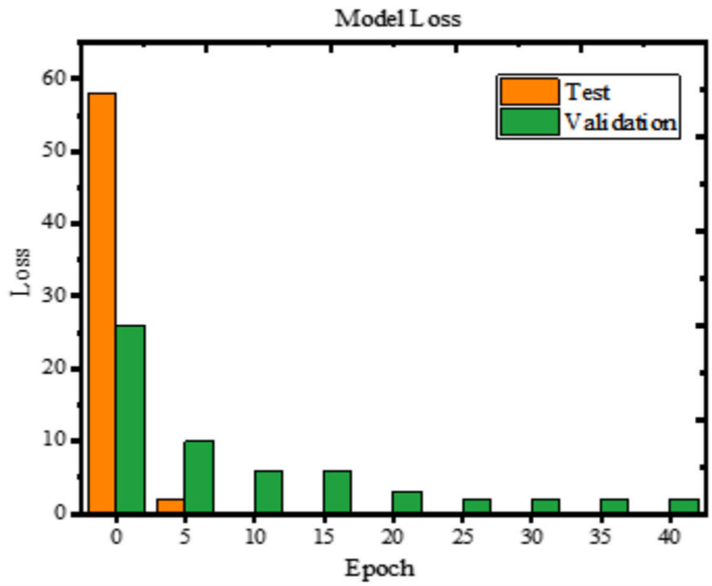

Fig. 6. Validation loss Vs loss plot for 40 epochs.

dataset and task. The Training Accuracy Vs Validation Accuracy Plot is depicted in Fig. 11. The Training Loss Vs Validation Loss Plot is depicted in Fig. 12.

4.3. Classification of tumor

The classification of tumors involves categorizing them into different types or classes based on various characteristics. This is a common task in medical imaging, where machine learning techniques are often used to assist in the automated classification of tumors in medical images. It's important to note that the success of tumor classification models often depends on the quality and diversity of the dataset, as well as the choice of features and model architecture. Regular collaboration with medical professionals is crucial to ensure the clinical relevance and accuracy of the model's predictions. Additionally, ethical considerations and regulatory guidelines should be followed when developing and deploying models for medical applications. The Confusion matrix, Accuracy plot and Loss plot for ResNet50 model are illustrated in Fig. 13, Fig. 14 and Fig. 15.

The Confusion matrix, Accuracy plot and Loss plot for EfficientNet model are illustrated in Fig. 16, Fig. 17 and Fig. 18. The Confusion Matrix Parameters for proposed CNN for various epochs are illustrated in Table 1.

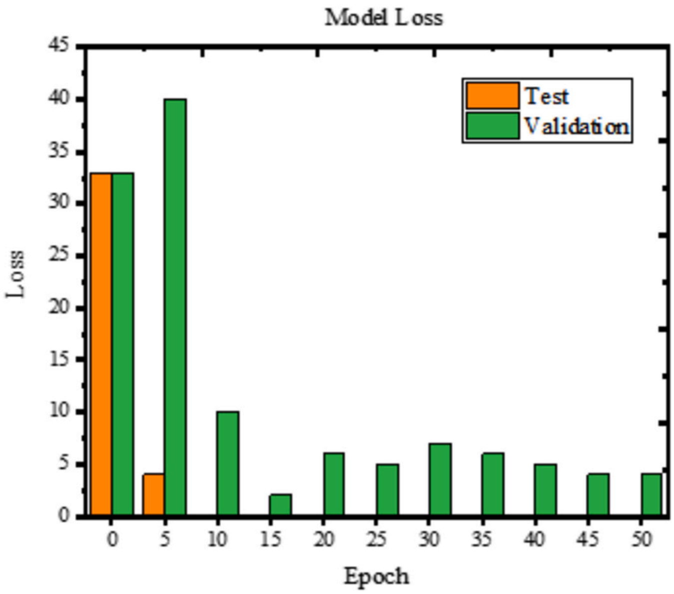

Fig. 7. Validation loss Vs loss plot for 50 epochs.

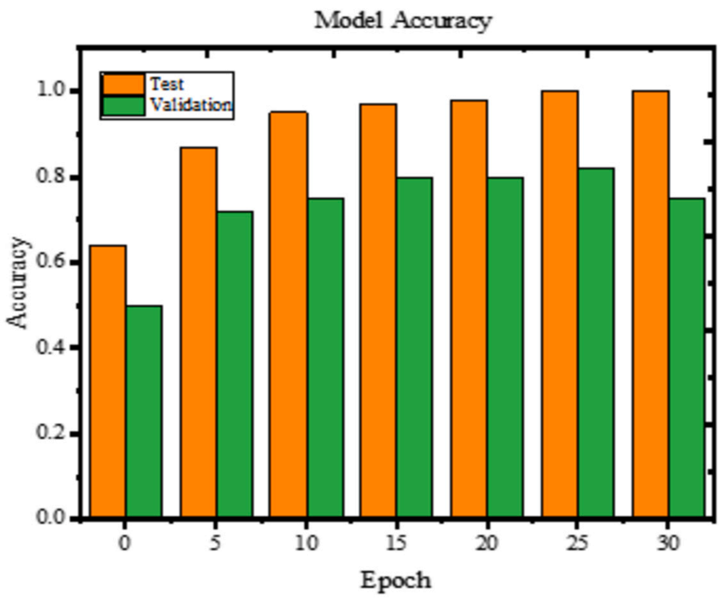

Fig. 8. Validation accuracy Vs training accuracy for 30 epochs.

Table 1
Confusion Matrix Parameters for proposed CNN.

	For 30 Epochs	For 40 Epochs	For 50 Epochs
True Positive (TP)	15	21	21
True Negative (TN)	15	12	11
False Positive (FP)	9	3	3
False Negative (FN)	3	6	7

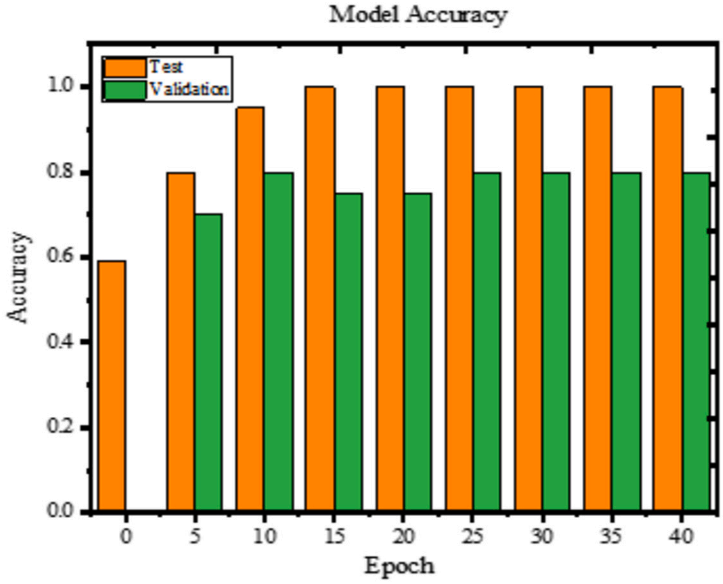

Fig. 9. Validation accuracy Vs training accuracy for 40 epochs.

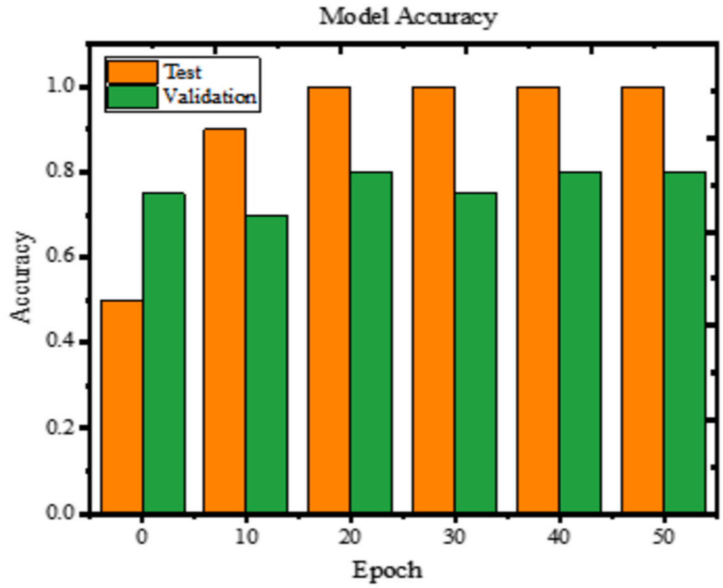

Fig. 10. Validation accuracy Vs training accuracy for 50 epochs.

4.4. Comparison of models

Comparison of parameters for various classes of brain images is depicted in Table-2. Model accuracy refers to the percentage of correct predictions made by a model on a given dataset. Model loss, also known as the loss function or cost function, quantifies the difference between the predicted values and the actual ground truth values. The relationship between model accuracy and model loss is typically inverse or opposite. Without augmentation the model achieves an accuracy of 97% in training and 71.3% in validation. However, when data augmentation is applied, these accuracies improve to 99% in training and 80% in validation. As the model accuracy increases, the model loss usually decreases, and vice versa. From the Table 2, we can observe that model experiences lower loss during training without augmentation compared to when augmentation is applied.

The Comparison of parameters for various classes of brain images of proposed methodology using ResNet50 and EfficientNet models are depicted in Table 3. The selection of criteria for evaluating a classification method is crucial to ensure the effectiveness of the proposed CNN method for detecting brain tumor. Traditionally, decision support systems have relied on information-gathering parameters for analysis. The comparison of accuracy, precision and recall of the models used for classification and detection of brain tumors is shown in Table 4. The proposed method achieves an accuracy of 96% in detecting brain tumors using ResNet50, while

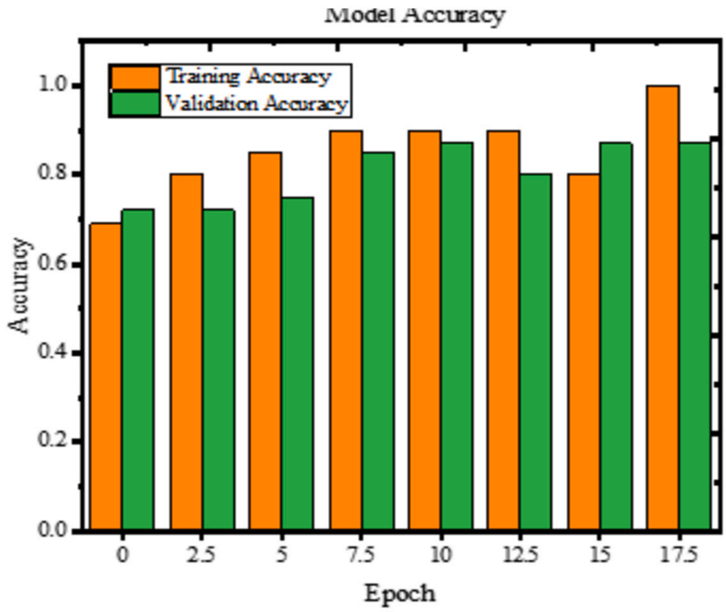

Fig. 11. Training Accuracy Vs Validation Accuracy Plot.

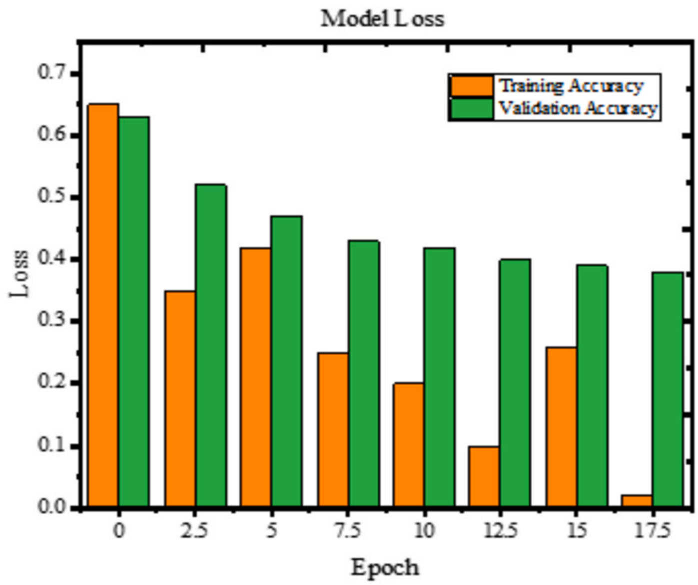

Fig. 12. Training Loss Vs Validation Loss Plot.

Table 2
Comparison of model accuracy and model loss for CNN with and without augmentation.

	Model Accuracy		Model Loss	
	Training	Validation	Training	Validation
Without augmentation	97%	71.3%	0.66 maximum 0.12 minimum	0.75 maximum 0.45 minimum
With augmentation	99%	80%	0.92 maximum 0.08 minimum	0.58 maximum 0.13 minimum

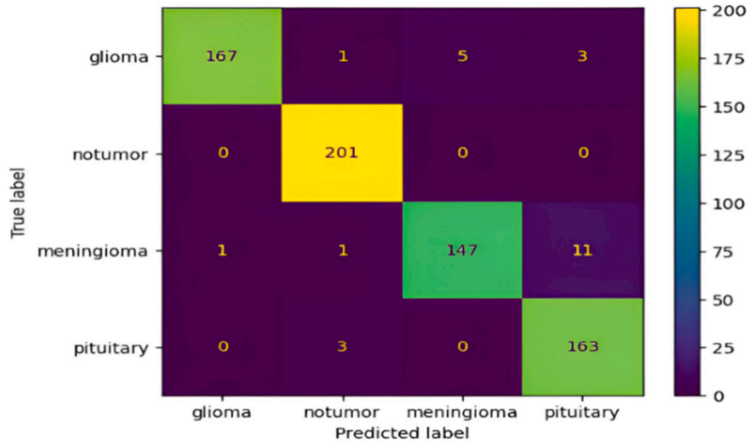

Fig. 13. Confusion matrix for ResNet50 model.

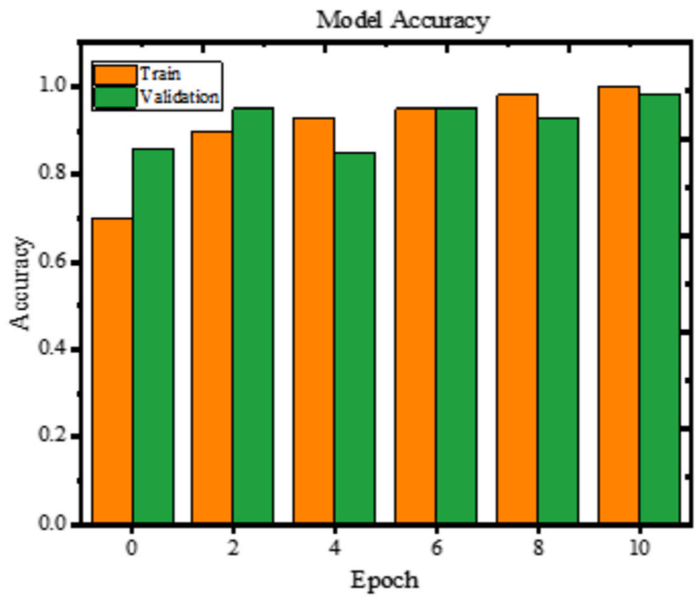

Fig. 14. Accuracy plot for ResNet50.

Table 3
Comparison of parameters for various classes of brain images.

	ResNet50				EfficientNet			
	TP	TN	FP	FN	TP	TN	FP	FN
GLIOMA	167	0	1	0	175	0	3	0
NOTUMOUR	201	0	5	0	200	0	3	0
MENINGIOMA	147	0	5	0	152	0	2	0
PITUITARY	163	0	4	0	162	0	6	0

utilizing EfficientNet yields a higher accuracy of 98%, as illustrated in Fig. 19. The proposed method achieves a precision of 97% in detecting brain tumors using ResNet50, while employing EfficientNet results in a higher accuracy of 98.5%, as illustrated in Fig. 20. The proposed method achieves a Recall of 98% in detecting brain tumors using ResNet50, while employing EfficientNet results in a higher accuracy of 99%, as illustrated in Fig. 21.

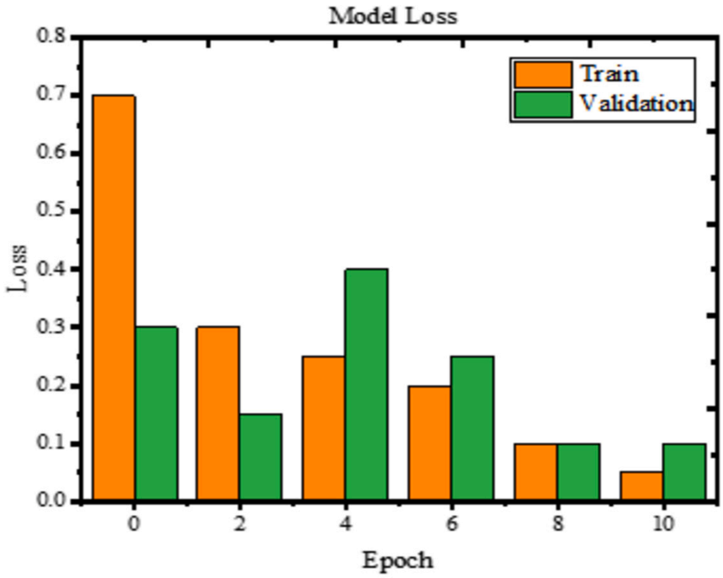

Fig. 15. Loss plot for ResNet50.

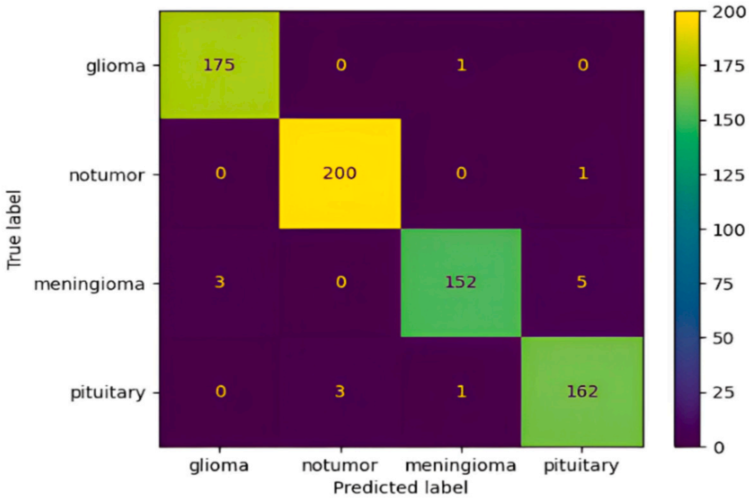

Fig. 16. Confusion matrix for EfficientNet.

Table 4
Comparison of Accuracy, Precision, and Recall Across Models in Brain Tumor Detection and Classification.

S.No	Model	Accuracy	Precision	Recall
1	CNN without data augmentation	78.57%	75%	87.5%
2	CNN with data augmentation	85%	82%	88%
3	ANN [10]	91.72%	89.2%	89.2%
4	ResNet50 (Proposed1)	96%	97%	98%
5	EfficientNet (Proposed2)	98%	98.5%	99%

5. Conclusion

The primary objective of this study is to employ deep learning for the detection and classification of brain tumors. Addressing challenges arising from a limited dataset and computing resources, we implemented transfer learning and image augmentation techniques. In the context of brain tumor detection, a Convolutional Neural Network (CNN) was utilized, achieving a validation accuracy

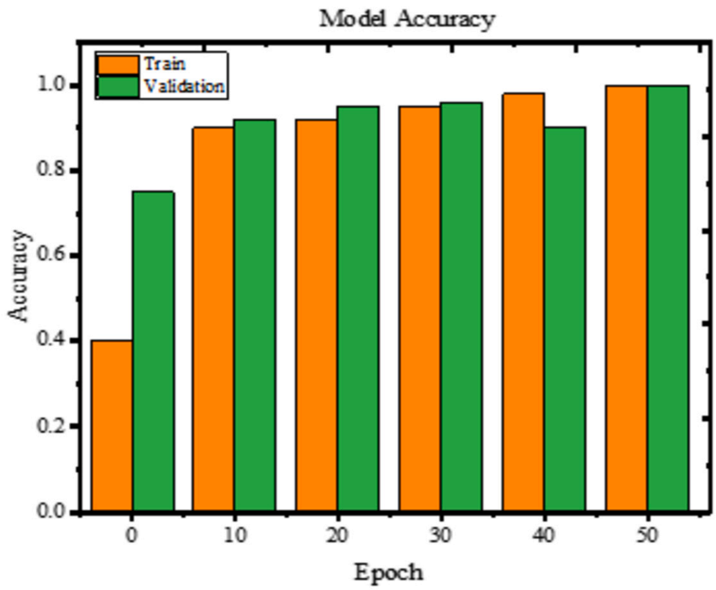

Fig. 17. Accuracy Plot for EfficientNet.

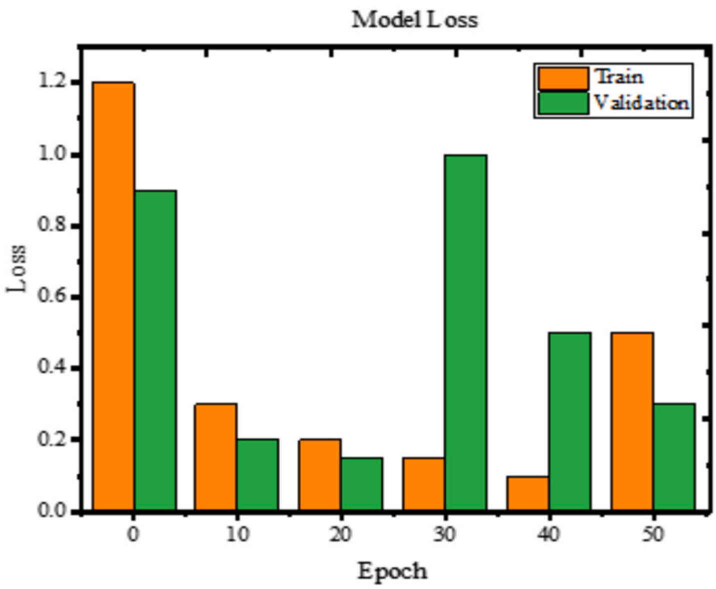

Fig. 18. Loss Plot for EfficientNet.

of 78.57% and a training accuracy of 97% without data augmentation. With data augmentation, the model demonstrated a validation accuracy of 85% and a training accuracy of 99%. For tumor classification, transfer learning models were employed. The ResNet50 model yielded a testing accuracy of 96% with a testing loss of 9.4%, precision of 97% and recall of 98%, while the EfficientNet model achieved a testing accuracy of 98% with a testing loss of 5.09%, precision of 98.5% and recall of 99%. This suggests the necessity of gathering more data for enhanced classification accuracy in future research. Furthermore, the exploration of alternative models and architectures beyond the employed fully connected architecture is recommended to ascertain the optimal approach for improved accuracy in brain tumor classification. Future investigations should involve the development and comparison of various models to inform the selection of the most effective architecture.

In future, it is imperative to gather additional data to augment classification accuracy and detection of brain tumor. Moreover, exploring alternative models and architectures beyond the currently employed fully connected architecture can be used to determine the most effective approach for enhancing accuracy in brain tumor classification.

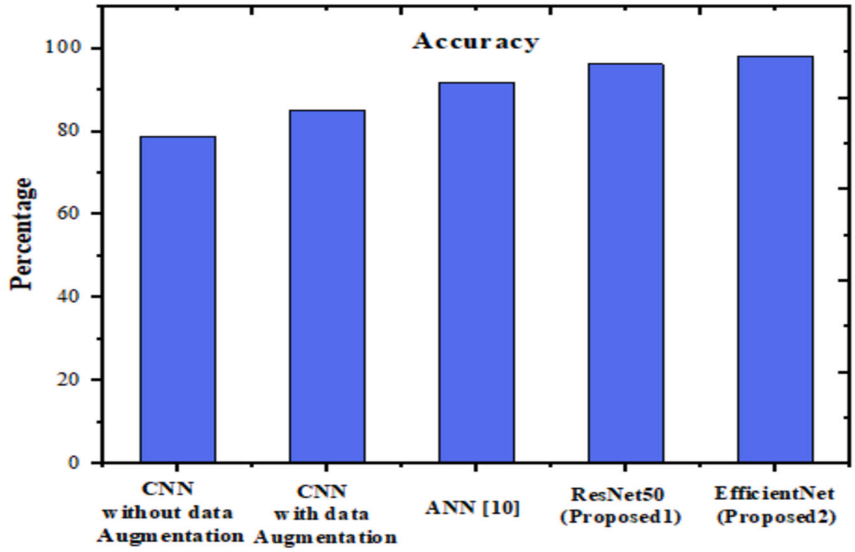

Fig. 19. Comparison of Accuracy of Proposed ResNet50 and EfficientNet models for detection of Brain tumor with existed works.

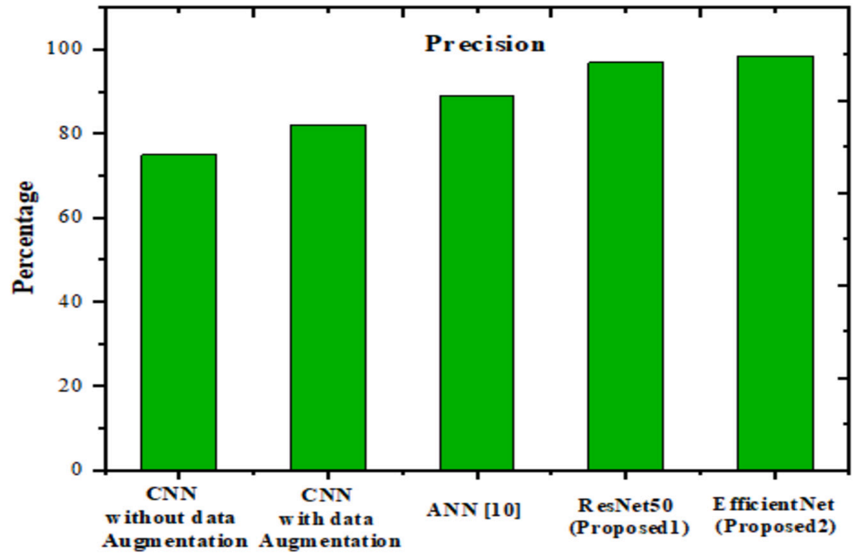

Fig. 20. Comparison of Precision of Proposed ResNet50 and EfficientNet models for detection of Brain tumor with existed works.

Funding

This research was funded by Princess Nourah Bint Abdulrahman University Researchers Supporting Project number (PNURSP2024R435), Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia.

CRediT authorship contribution statement

K. Nishanth Rao: Resources, Formal analysis. **Osamah Ibrahim Khalaf:** Visualization, Investigation, Data curation. **V. Krishnasree:** Project administration, Methodology, Conceptualization. **Aruru Sai Kumar:** Writing – review & editing, Validation, Software, Data curation. **Deema Mohammed Alsekait:** Supervision, Investigation, Funding acquisition. **S. Siva Priyanka:** Validation, Formal analysis. **Ahmed Saleh Alattas:** Visualization, Funding acquisition. **Diaa Salama AbdElminaam:** Validation, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

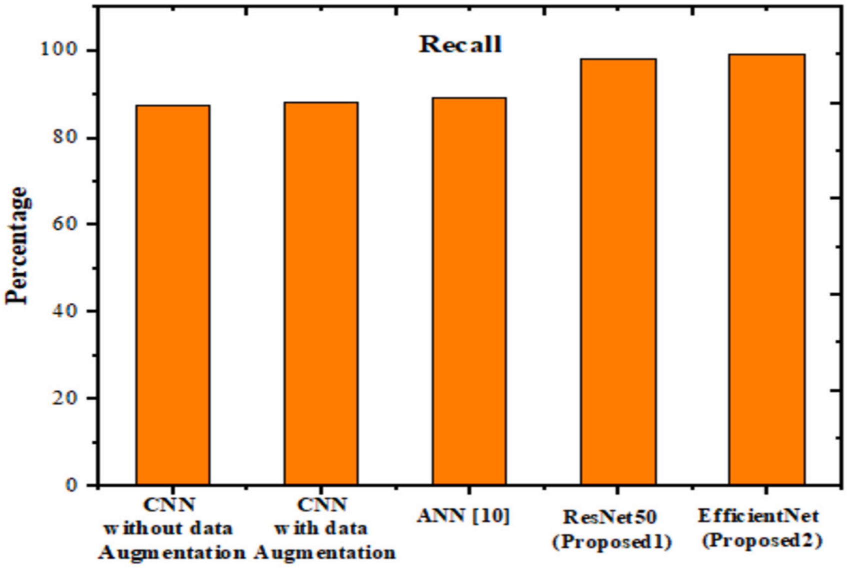

Fig. 21. Comparison of Recall of Proposed ResNet50 and EfficientNet models for detection of Brain tumor with existed works.

Data availability

Data will be made available on request.

Acknowledgement

The authors would like to acknowledge the support of Princess Nourah Bint Abdulrahman University Researchers Supporting Project number (PNURSP2024R435), Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia.

References

[1] D.R. Nayak, N. Padhy, P.K. Mallick, M. Zymbler, S. Kumar, Brain tumor classification using dense efficient-net, Axioms 11 (2022) 34.
[2] Pankaj Sapra, Rupinderpal Singh, Shivani Khurana, Brain tumor detection using neural network, Int. J. Sci. Mod. Eng., IJISME (ISSN 2319-6386) 1 (9) (2013).
[3] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556, 2014.
[4] V.R. Sajja, Classification of brain tumors using fuzzy C-means and VGG16, Turk. J. Comput. Math. Educ. (TURCOMAT) 12 (2021) 2103–2113.
[5] K.G. Khambhata, S.K. Panchal, Multiclass classification of brain tumor in MR images, Int. J. Innov. Res. Comput. Commun. Eng. 4 (5) (2016) 8982–8992.
[6] J. Seetha, S. Selvakumar Raja, Brain tumor classification using convolutional neural networks, Biomed. Pharmacol. J. 11 (3) (2018).
[7] Pankaj Sapra, Rupinderpal Singh, Shivani Khurana, Brain tumor detection using neural network, Int. J. Sci. Mod. Eng., IJISME (ISSN 2319-6386) 1 (9) (August 2013).
[8] P. Gokila Brindha, Brain tumor detection from MRI images using deep learning techniques, in: IOP Conference Series: Materials Science and Engineering, 2020.
[9] M. Aarthilakshmi, S. Meenakshi, A. PoornaPushkala, V. Rama@Ramalakshmi, N.B. Prakash, Brain tumor detection using machine learning, Int. J. Sci. Technol. Res. 9 (04) (2020).
[10] Dheiver Santos, Ewerton Santos, Brain Tumor Detection Using Deep Learning, medrxiv for heath services, January 25, 2022.
[11] Javaria Amin, Muhammad Sharif, Anandakumar Haldorai, Mussarat Yasmin, Ramesh Sundar Nayak, Brain tumor detection and classification using machine learning: a comprehensive survey, Complex Intell. Syst. 8 (23) (2022) 3161–3183.
[12] Sonali B. Gaikwad, Madhuri S. Joshi, Brain tumor classification using principal component analysis and probabilistic neural network, Int. J. Comput. Appl. (ISSN 0975-8887) 120 (3) (2015).
[13] Raheleh Hashemzehi, Seyyed Javad Seyyed Mahdavi, Maryam Kheirabadi, Seyed Reza Kame, Detection of brain tumors from MRI images base on deep learning using hybrid model CNN and NADE, Biocybern. Biomed. Eng. (2020).
[14] Tanzila Saba, Ahmed Sameh Mohamed, Mohammad El-Affendi, Javeria Amin, Muhammad Sharif, Brain tumor detection using fusion of hand crafted and deep learning features, Cogn. Syst. Res. 59 (2020) 221–230.
[15] Sharan Kumar, Dattatreya P. Mankame, Optimization driven deep convolution neural network for brain tumor classification, Biocybern. Biomed. Eng. 40 (3) (2020) 1190–1204.
[16] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A. Courville, Y. Bengio, C. Pal, P.M. Jodoin, H. Larochelle, Brain tumor segmentation with deep neural networks, Med. Image Anal. 35 (2017) 18–31.
[17] S. Hussain, S.M. Anwar, M. Majid, Segmentation of glioma tumors in brain using deep convolutional neural network, Neurocomputing 282 (2018) 248–261.
[18] G. Litjens, T. Kooi, B.E. Bejnordi, A.A.A. Setio, F. Ciompi, M. Ghafoorian, J.A. Van Der Laak, B. Van Ginneken, C.I. Sánchez, A survey on deep learning in medical image analysis, Med. Image Anal. 42 (2017) 60–88.
[19] P. Chinnasamy, Wing-Keung Wong, A. Ambeth Raja, Osamah Ibrahim Khalaf, Ajmeera Kiran, J. Chinna Babu, Health recommendation system using deep learning-based collaborative filtering, Heliyon 9 (12) (2023).
[20] Aruru Sai Kumar, T.V.K. Hanumantha Rao, Scalable benchmark synthesis for performance evaluation of NoC core mapping, Microprocess. Microsyst. 79 (November 2020).
[21] Daniel Reyes, Javier Sánchez, Performance of convolutional neural networks for the classification of brain tumors using magnetic resonance imaging, Heliyon 10 (3) (2024).

[22] Aruru Sai Kumar, T.V.K. Hanumantha Rao, An adaptive core mapping algorithm on NoC for future heterogeneous system-on-chip, Comput. Electr. Eng. 95 (October 2021).
[23] A.S. Kumar, et al., Performance and communication energy constrained embedded benchmark for fault tolerant core mapping onto NoC architectures, Int. J. Ad Hoc Ubiq. Comput. 41 (2) (2022) 108–117.
[24] Lu Xu, Morteza Mohammadi, Brain tumor diagnosis from MRI based on Mobilenetv2 optimized by contracted fox optimization algorithm, Heliyon 10 (1) (2024).
[25] A.S. Kumar, B. Naresh Kumar Reddy, An efficient real-time embedded application mapping for NoC based multiprocessor system on chip, Wirel. Pers. Commun. 128 (2023) 2937–2952.
[26] S. Talasila, K. Rawal, G. Sethi, Black gram disease classification using a novel deep convolutional neural network, Multimed. Tools Appl. 82 (2023) 44309–44333.
[27] S. Pereira, A. Pinto, V. Alves, C.A. Silva, Brain tumor segmentation using convolutional neural networks in MRI images, IEEE Trans. Med. Imaging 35 (5) (2016) 1240–1251.
[28] S. Talasila, K. Rawal, G. Sethi, PLRSNet: a semantic segmentation network for segmenting plant leaf region under complex background, Int. J. Intell. Unmanned Syst. 11 (1) (2021) 132–150.
[29] S. Ganesh, S. Kannadhasan, A. Jayachandran, Multi class robust brain tumor with hybrid classification using DTA algorithm, Heliyon 10 (1) (2024).
[30] J. Kang, Z. Ullah, J. Gwak, MRI-based brain tumor classification using ensemble of deep features and machine learning classifiers, Sensors 21 (6) (2021).
[31] K. Sarangam, A.S. Kumar, B.N.K. Reddy, Design and investigation of the 22 nm FinFET based dynamic latched comparator for low power applications, Trans. Electr. Electron. Mater. (2024) 1–14.
[32] B.N.K. Reddy, A.S. Kumar, Evaluating the effectiveness of bat optimization in an adaptive and energy-efficient network-on-chip routing framework, J. Parallel Distrib. Comput. (2024) 104853.