

An Efficient Approach for Brain Tumor Detection Using Deep Learning Techniques



R. V. Belfin, J. Anitha, Aishwarya Nainan, and Lycia Thomas

Abstract In today's biomedical scenario, medical automation is improvising and is in demand on an extensive scale. Detecting brain tumor also needs automation for further treatment. There are many brain imaging methods like X-Ray, CT scans, MRI, EEG, and PET. The main aim of these medical imaging techniques is to extract all meaningful and precise information from images with the least error possible. MRI is a non-invasive technique that is currently used for brain analysis, where an experienced doctor has to manually and precisely identify brain tumors in the MRI scans. This paper presents computer-aided deep learning approaches to detect and segment the brain tumor from the MRI images taken from the standard dataset. The proposed work focuses on three main steps like preprocessing of the images, classifying data into the tumor and non-tumorous images, and segmenting the brain tumor region. Convolutional Neural Network (CNN) is a deep learning technique that is mainly used in image classification. In this work, a VGG16 CNN architecture is utilized to classify brain images. Further, computer vision is used to semantically segment the tumorous areas in the MRI scans. The weights of the trained CNN VGG-16 model are used to build bounding boxes and to segment the tumor regions of the MRI scans using the Mask R-CNN model. Mask R-CNN model is a widely used computer vision algorithm that segments and highlights the tumor region in the MRI scan. Thus, using CNN and computer vision techniques, the proposed approach can able to achieve a highly precise medical automation system that can detect and identify brain tumors with improved precision.

Keywords Brain tumor · MRI scans · Convolution neural network · Mask R-CNN

R. V. Belfin (✉) · J. Anitha · A. Nainan · L. Thomas
Karunya Institute of Technology and Sciences, Coimbatore, India
e-mail: researchbelfin@gmail.com

J. Anitha
e-mail: anitha_j@karunya.edu

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
A. Khanna et al. (eds.), *International Conference on Innovative Computing and Communications*, Advances in Intelligent Systems and Computing 1387,
https://doi.org/10.1007/978-981-16-2594-7_25

297

1 Introduction

A brain tumor is the abnormal growth of cancer cells in the brain. Malignant and Benign are two types of tumors present in the brain. Malignant tumors are cancerous as they spread rapidly into cells via the bloodstream or lymphatic system. These cells consist of DNA and abnormal chromosomes. It can create fatigue and weight loss. This tumor requires aggressive treatment including surgery, chemotherapy, and radiation. Benign tumors are not cancerous as the cells do not tend to spread because the cell's growth is slower. This tumor usually doesn't recur once removed. It is believed that early detection of tumors before their rapid progress is very crucial to save lives. Survival depends on what stage the person is diagnosed and the immediate medical response to the diagnosis. Thus, in order to prevent such uncertainty to occur, a great way is to automate the entire inferential treatment on a highly precise automated system that detects the tumor in a very early stage. To have such highly precise automation, there must be a need for developing computer automated algorithms that don't rely much on features manually extracted or any hard-coded external factors. Hence, the use of convolutional neural networks with the additional assistance of transfer learning techniques enables us to build such a highly precise and robust automation system. CNN is one of the most powerful and widely used algorithms when it comes to the sub-field of machine learning called deep learning. One of its key features is learning key insights and parameters in data on its own. A convolutional neural network on its own learns specific insights on how objects look like. Thus CNN is widely beneficial in the medical field as it is able to learn, depict, and predict complex medical MRI analysis on its own provided with a good amount of data.

2 Related Works

Extensive research works have been done in building highly precise brain tumor detection systems in recent years.

The authors of [9] have worked on the fetus MRI images to detect the early stage of brain development. They have used CNN to perform the segmentation process. The metrics such as Dice coefficient (DC) and Mean Surface Distance (MSD) are used to annotate various surfaces. Charron et al. [4] have worked on the segmentation of gliomas using 3D CNN. They have used network parameters to analyze the brain and evaluated the network performance in terms of detection and segmentation. Moeskops et al. [13] have used CNN to segment the tissues in each voxel of the brain image. The average dice coefficient is used to evaluate their segmentation method. They have demonstrated robustness in the various acquisition and age difference in the target MRI images.

Some of the literature like [2, 3, 6, 10, 11, 15] has recently worked in brain tumor images by using different variants in the CNN algorithm to segment and classify the tumorous voxels. Few other authors like [8, 17, 19–21] have explored the MRI images with various methods from evolutionary algorithms and deep learning techniques. In [8], the authors have suggested techniques for classification and segmentation of MRI images of the brain. They have classified the dataset into two types of tumor, i.e., glioma and meningioma. Classification is done using a convolutional neural network and segmentation is performed using the Faster R-CNN. The t1-weighted brain tumor dataset is taken in this work. The system presented an average accuracy of 94% for all classifiers.

The approach proposed by [7] uses prepossessing, classification, and segmentation. The dataset includes three kinds of brain tumors, i.e., meningioma, glioma, and pituitary tumor. Prepossessing is done by methods like filtering using the Gaussian filter and equalization technique. The classification is performed using Convolutional Neural Network along with Adam optimizer for getting the best optimization results. The accuracy obtained is 94.39%. In [19], the authors have used Convolutional Neural Network along with three classifiers namely Softmax, Radial Basis Function (RBF), and Decision Tree (DT). The accuracy of the CNN with the Softmax classifier is 98.67%. Also, the accuracy of CNN with RBF kernel classifier and DT classifier is 97.34% and 94.24%, respectively. In the approach of [18], methods like preprocessing, image enhancement using histograms, feature extraction, and segmentation have been used. This proposed technique helps to obtain accurate automatic segmentation and tumor detection with the help of CNN and spatial domain techniques. In the approach of [12], the authors have proposed a system to perform segmentation and classification using a Fully Connected Neural Network (FCNN). The two main functions performed are pre-dispensation and characteristic removal. Initially, the images are preprocessed by applying a filter to the images. Then, the proposed method uses the FCNN with which the semantic segmentation is performed.

Parihar et al. [14] have done a study on CNN-based MRI segmentation. A comparative analysis has been done between various brain tumor segmentation using CNN. In [16], automatic segmentation is proposed based on CNN by exploring small 3*3 kernels in the Brain Tumor Segmentation Challenge 2013 database (BRATS 2013). Further, the use of intensity normalization in the preprocessing step and the importance of data augmentation in brain tumor segmentation are investigated. While in [13], automatic segmentation is performed on MR brain images using CNN. The evaluation has been done with the different datasets of images acquired at distinct ages.

3 Deep Learning

Deep learning is one of the fundamental advances in the field of artificial intelligence. Deep learning comprises high computationally performing machine learning algorithms called neural networks that mimic the functioning of neurons in the human

brain. These neural networks are a densely interconnected network of nodes interconnected with each other, where each node holds and represents a computation. These groups of interconnected neurons together forward each layer-wise computations to further preceding neurons and thus all the dense networks of neurons are responsible for building a learning mechanism inside a neural network architecture. As the network becomes deeper and wider, it develops more complex problems solving ability. This state of behavior represented by a deep dense interconnected networks of neurons is called deep learning. Neural networks are heavily data and computational driven, which enables them to yield higher accuracy and higher performance metric scores as compared to any previous versions of standard machine learning algorithms. Neural networks also do all the unbound statistics and feature engineering all the data points on their own internally. It happens because of the densely interconnected network that on its own distributed learning mechanism across nodes and layers in the network to propagate throughout the computation. This wide and densely connected architecture enables neural networks to be used across a wide variety of problem statements and has made it a wide option for performing highly precise machine learning computations across all domains. The architecture diagram and the flowchart of the proposed system are shown in Fig. 1 and Fig. 2, respectively.

3.1 Convolutional Neural Network

Convolutional Neural Network is a special kind of neural network derived from the sparse connection of networks that are widely used on images. The problem with standard normal neural networks is that it overfits very quickly when it comes to learning from image data because the representation of data in images is quite different than data represented in the form of data points. Convolutional Neural Networks work on two important factors such as shared computations and sparsity in connections. Shared computations mean the networks share their learning from a particular region of the image to various other blocks of nodes. Sparse computations mean the computations occurring on the data have sparsity that is a specific area of a region of the image is picked and computations are inferred on that region of that part of the image. These two specific features in CNN architecture enable the neural network to have a good amount of computation to work on images while having a definite number of learning parameters and preventing the model from overfitting. The CNN architecture consists of various different aspects of layers each distinct and fulfilling a different premise in the landscape of Convolutional Neural Network architecture. A CNN consists of layers like convolutional layer, pooling layer, dense layers, flatten layer, and output layer along with various other hyperparameters like padding, strides, sparsity, etc.

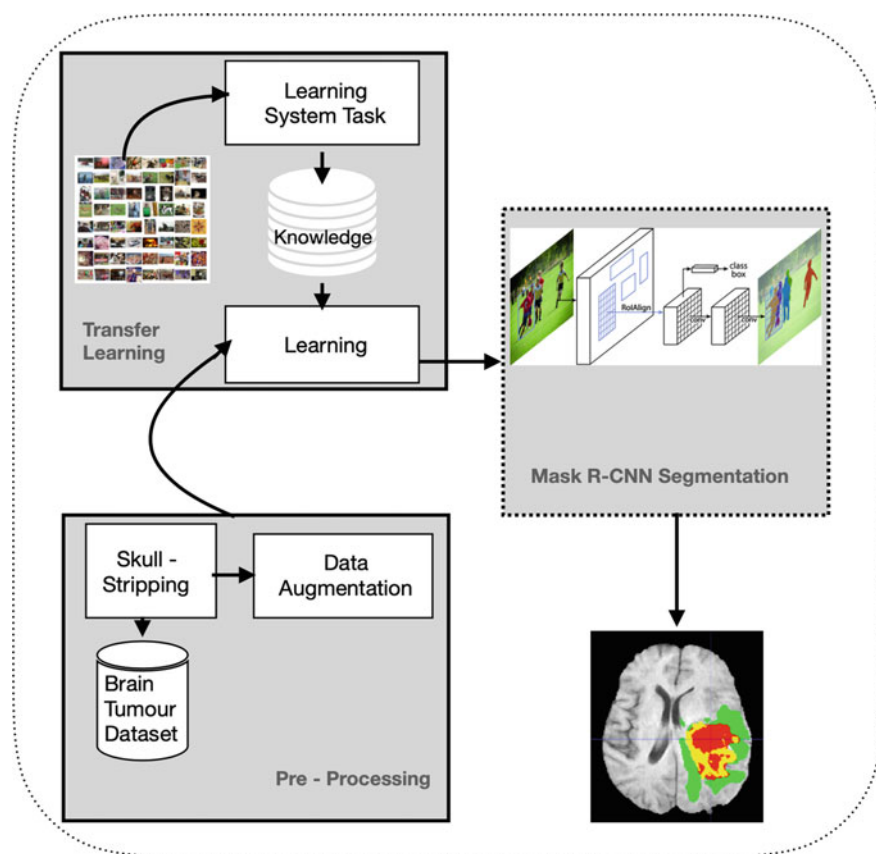
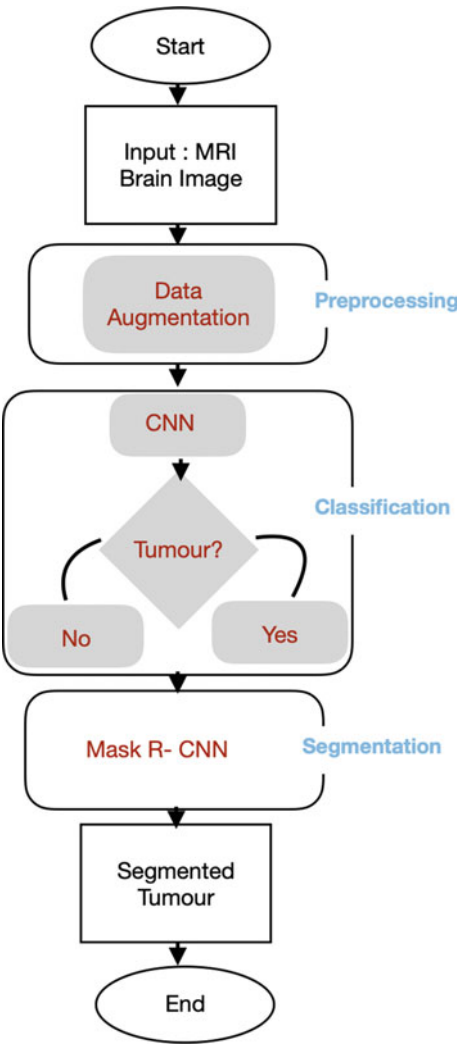


Fig. 1 Architecture of proposed system

3.2 Convolutional Layer

The convolutional layer is one of the first and fundamental layers of CNN. This layer is responsible for proving the network does sparsity in computations. The convolutional layer processes each area of the image iteratively, where an area comprises of a few matrices of pixels over a wider image. These computations emphasize the learning it can achieve in that layer of the image area. Convolutional layers have filters that slide over such regions of the image called receptive fields. The sliding of the filter over the image is called convolving or a convolutional mathematical matrix operation that occurs between the matrix values of the filter and the matrix value representing the region of the image. The extracted computations from the convolutional operations yield weights and learning inferred from the images. It mathematically infers what's present inside the image. The beginning first few layers of the convolutional operation of the network absorb horizontal and vertical edges of the images following which

Fig. 2 Flowchart of the proposed system



the further layers capture small objects to much bigger objects and hence building the entire learning mechanism from images.

3.3 Pooling Layer

A pooling layer is a layer that is added to the network after every layer of the convolutional layer with the activation function ReLu. The pooling layer utilizes a downsampling methodology, where it downsamples feature maps by summarizing

the presence of features in patches of the features in images. The most common operations in pooling layers are max pooling and global max pooling which yield max value or global average value from the convolutional layers of the feature maps and grasp only the important features from the feature maps. The pooling layers provide an abstracted representation of features from the convolutional layers.

3.4 Flatten and Dense Layer

Flatten and fully connected dense layers are the last stages of layers in the CNN architecture. These layers are the layers that produce the predictive output from the network. The flatten layer flattens out the last convolutional layer into dense 1xn layers of neurons. These layers make a dense connection with neurons from the previous convolutional layer. The fully connected dense layer consists of a certain amount of neurons that further networked with the output layer over a sigmoid or softmax activation function to predict the output classes. In this work, a sigmoid output function is used for the binary classification problem.

3.5 Strides

The strides is a hyperparameter feature that depicts how much units of padding the convolutional filter needs to shift in each convolutional operation. Most commonly strides are set to one. But to yield faster results on the convolutional operation and to prevent data noise, higher values of strides are also used, which makes the convolutional operations faster.

3.6 Padding

Padding is used to maintain the same size of the feature as the input image after the convolution. Padding is used to increase the dimensions by adding additional pads over the input data and also in the preceding layers before a convolutional operation occurs. Padding increases the data dimensions so that convolutional operation takes lesser amounts of shrinking. This enables us to highlight the border pixels and reduce the loss of information while also yielding better learning from various convolutional layers.

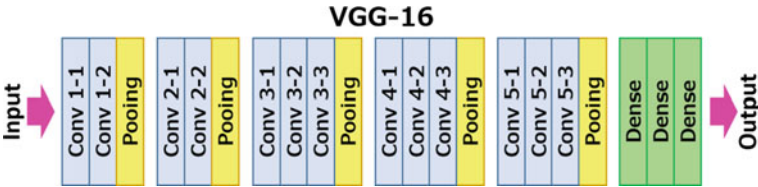


Fig. 3 VGG-16 model [1]

4 CNN Architectural Model

Figure 3 shows the VGG-16 model used in this work, which is one of the widely used CNN architectures because of its distinct architectural design. VGG-16 has smaller convolutional 3x3 filters with a stride of 1 and a max-pooling filter with a stride of 2. VGG-16 has 16 layers of architecture that have weights comprising 138 million learning parameters. Additionally, the dimensions increase from 64, 128, 256 to 512, and this enables the network architecture to follow a uniform computation and produces much higher accuracy results. VGG-16 is used with transfer learning methodologies that enable the use of pre-trained weights from ImageNet. The transfer learning concept enables the network architecture to be fine-tuned on a finer and specific area of distinct problem domains.

5 Data

The dataset [5] collected in this work consists of 253 total images with 155 images of patients having cancer and 98 images of patients not having cancer. The data consists of T2 MRI scans of the brain. The T2 strip MRI scan of the brain exposes the entire cerebral cortex of the human brain thus exposing the most prominent area where the tumor cells clog and coagulate. Thus these scanned images are vital in developing the prediction system. Further, the input dataset is split into 193 images for training, 50 images of validation, and 10 images for testing.

5.1 Data Augmentation

The input dataset consisting of 253 images is augmented with a data augmentation function to increase the diversity of the input data. Data augmentation further multiplies the data several times where the actual input data has minor deviations in the visual aspects of the data. The parameters used for data augmentation are width shift, rotation range, height shift, re-scale, brightness range, horizontal flip, and vertical flip. Figure 4 shows the augmented data for a sample image from the dataset.

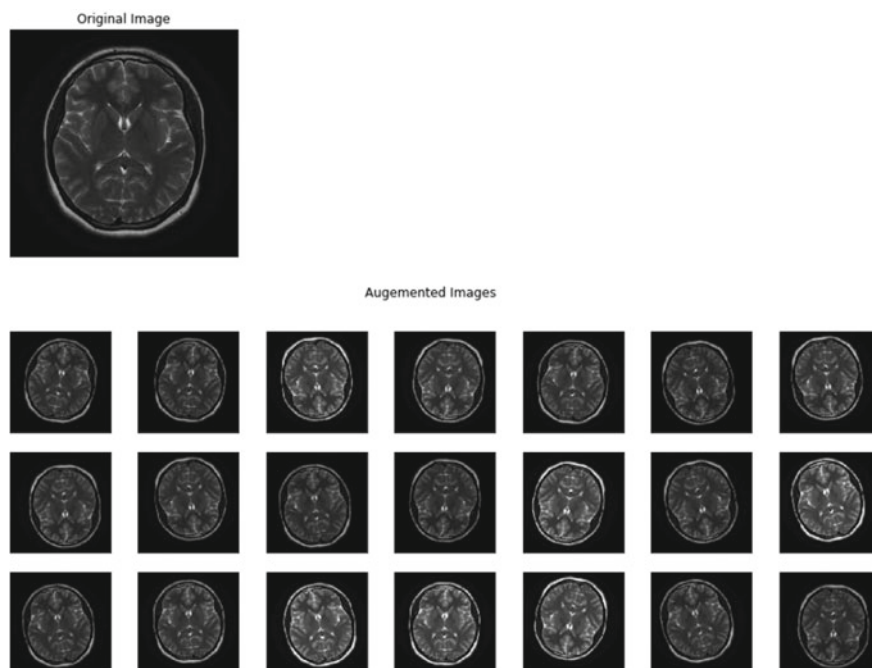


Fig. 4 Data Augmentation

6 Computer Vision

Computer vision abstracts the learning from the CNN images and generates more detailed, specific, and spontaneous results from the neural network. Computer vision techniques enable building bounding boxes around image data to predict various segments and objects in images. Computer vision techniques are extensively used for image segmentation, object tracking, and localization. The underlying computations in computer vision utilize a Recurrent Convolutional Neural Network (R-CNN) architecture that builds up a sliding window mechanism on different sizes of receptive fields. This enables autonomous systems, autonomous image tracking, and localization systems at scale. This work utilizes a quite common Mask R-CNN computer vision technique to extract the tumor cells region in the T2 scan MRI images. The mask is a visual segmentation of the tumor with an image highlighting on a localized border, along with bounding boxes predicting the confidence values of the particular region in a tumor cancer cell region.

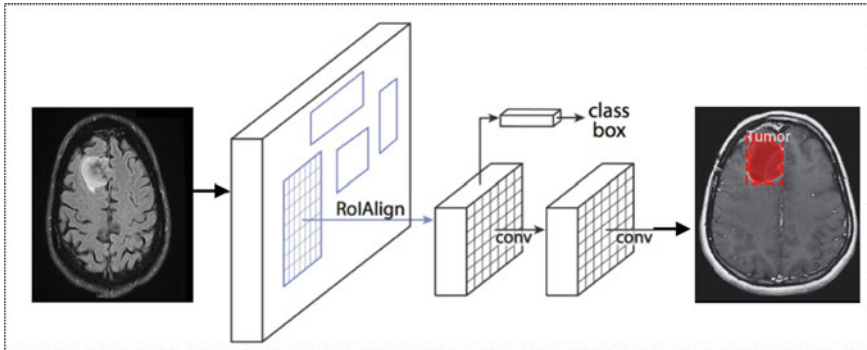


Fig. 5 Mask R-CNN

6.1 Mask R-CNN

The mask R-CNN extends the faster R-CNN architectural approach which is shown in Fig. 5. It detects objects in the image while simultaneously generating a segmented region of the mask on the object for each instance. The mask R-CNN builds upon the COCO image data corpus. In this work, the mask R-CNN model uses the previously trained weights of the VGG16 architecture as a base learning and further learns to semantically build bounding mask segments over the object regions. Further, the model is trained and fine-tuned on the brain tumor MRI scanned images. The architecture consists of ResNet101 as a backbone architecture with backbone strides of [4, 8, 16, 32]. The model is trained for 15 epochs of 100 batches per epochs. The parameters set for the experiment classification are the minimum confidence value of 0.85, a learning rate of 0.01, a learning momentum of 0.9, and weight decay of 0.0001 with 50 validation steps.

7 Model Implementation

The VGG16 model is used as a base layer with all of its initial layers frozen except for the last two sets of Convolutional 2D and pooling 2D layers. Further, custom output layers are developed that include global average pooling 2D layer that picks the global average values from the matrix grid of the receptive regions. The dense layer of 256 neurons is added with the ReLu activation function, following by a dropout of 0.5 is added to prevent the model from overfitting. Finally, the output layer consists of two neurons with a sigmoid activation function to predict the output as tumor or non-tumor. The model is set to use batch gradient descent with 5 steps per epochs weight updates. The training batch size is 100 and the validation batch size is 10. The optimizer used is Adam and the loss function used is binary cross-entropy with a learning rate of 0.001 and a decay value of “1e-3 / 20” as shown in Fig. 6.

Layer (type)	Output Shape	Param #
vgg16 (Model)	(None, 7, 7, 512)	14714688
global_average_pooling2d_1 (	(None, 512)	0
dense_1 (Dense)	(None, 256)	131328
dropout_1 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 2)	514
Total params: 14,846,530		
Trainable params: 2,491,650		
Non-trainable params: 12,354,880		

Fig. 6 Model Architecture

Table 1 Performance metrics

	Precision	Recall	F1-score	Support
NO	0.85	0.89	0.87	19
YES	0.93	0.90	0.92	31
Macro avg	0.89	0.90	0.89	50
Weighted avg	0.90	0.90	0.90	50
Accuracy			0.90	50

8 Model Output

The dataset is split into the tumor and non-tumor images as shown in Fig. 7. The model is trained for 200 epochs with 5 steps per epoch weight updates. On each update, the model results in metrics such as training accuracy, validation accuracy, training loss, and validation loss as shown in Fig. 8. The model on training yields a 90 percent accuracy rate, with the following performance metrics of the classification report. The classification report consists of the precision, recall, and f1-score values also holding class-specific values for the same as shown in Table 1. Further, the classified tumor images are used with the R-CNN model to segment the brain tumor areas with the bounding boxes. The output of the segmented boundary of brain tumor images with the ground truth is shown in Fig. 9.

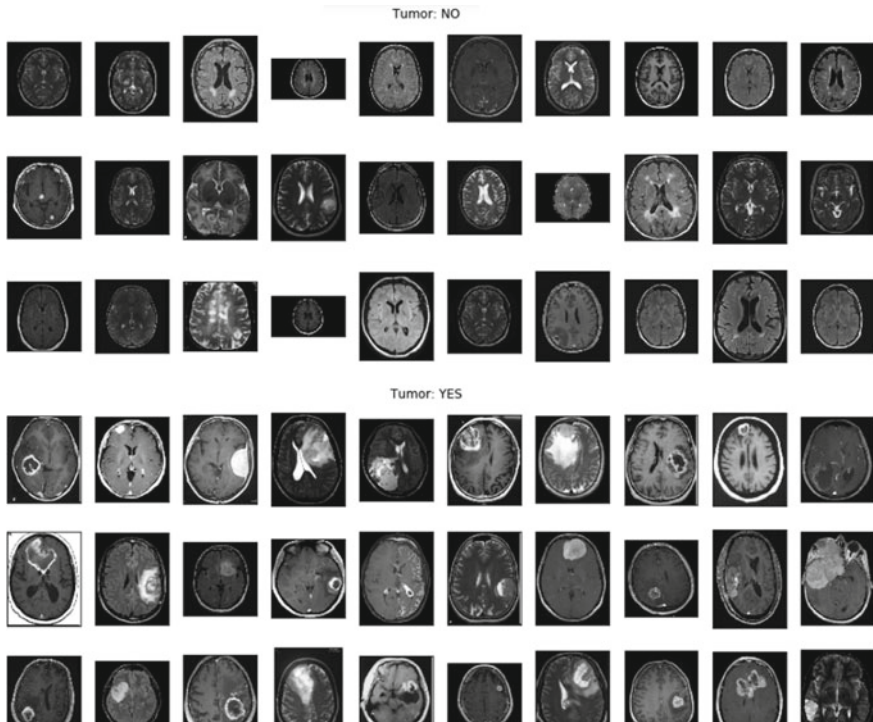


Fig. 7 Dataset splitted into tumor and non-tumor images

9 Conclusion

In this study, a powerful approach that can detect and segment brain tumors has been proposed. Initially, preprocessing steps like data normalization and data augmentation are performed in order to get better results. In the proposed approach, VGG16 CNN architecture with transfer learning method is used to initially classify the brain images into tumor and non-tumor. This architecture uses a transfer learning method where the initial weights of the model are frozen as the model is loaded pre-trained on the ImageNet weights. Further, the segmentation of the MRI image is performed to segment the tumor region from the image. An algorithm called Mask R-CNN is used to determine the area where the tumor is present in the brain. Also, the exacted region of the tumor is highlighted. The Mask R-CNN Is also pre-trained on the COCO dataset that is a common object classification dataset, where the model learns to generalize on the segmentation task. The VGG-16 model weights are used to further fine-tune the Mask R-CNN model to work with the MRI images to segment the tumorous region. The VGG16 CNN model gave an accuracy of 90% in classification. The accuracy, precision, and computational speed of segmentation methods will be further improved in our future research work.

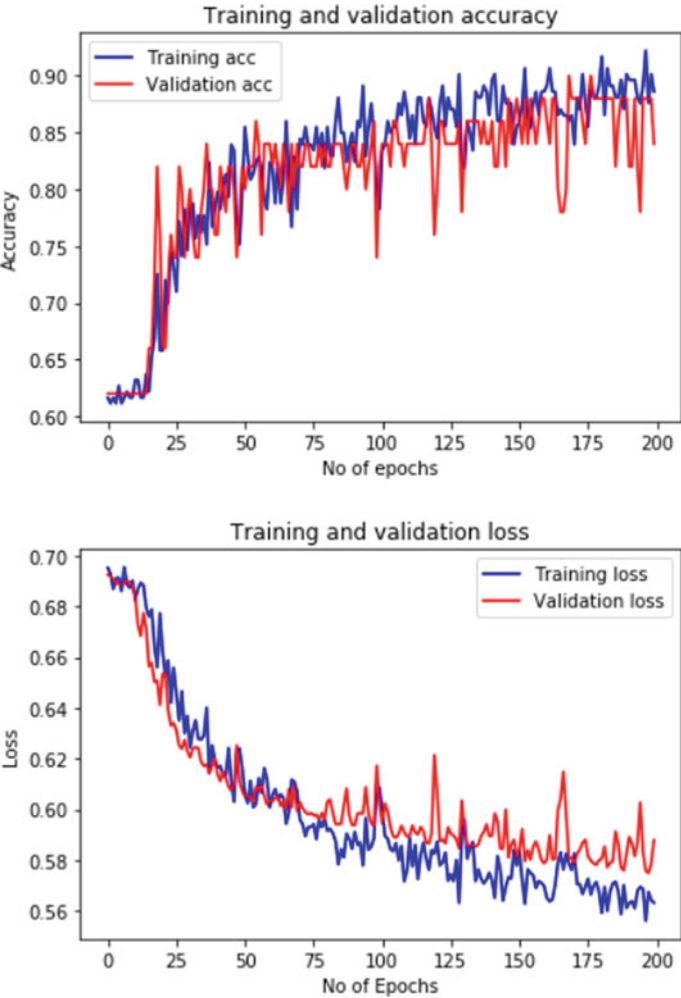


Fig. 8 Accuracy and loss of Testing and Validation

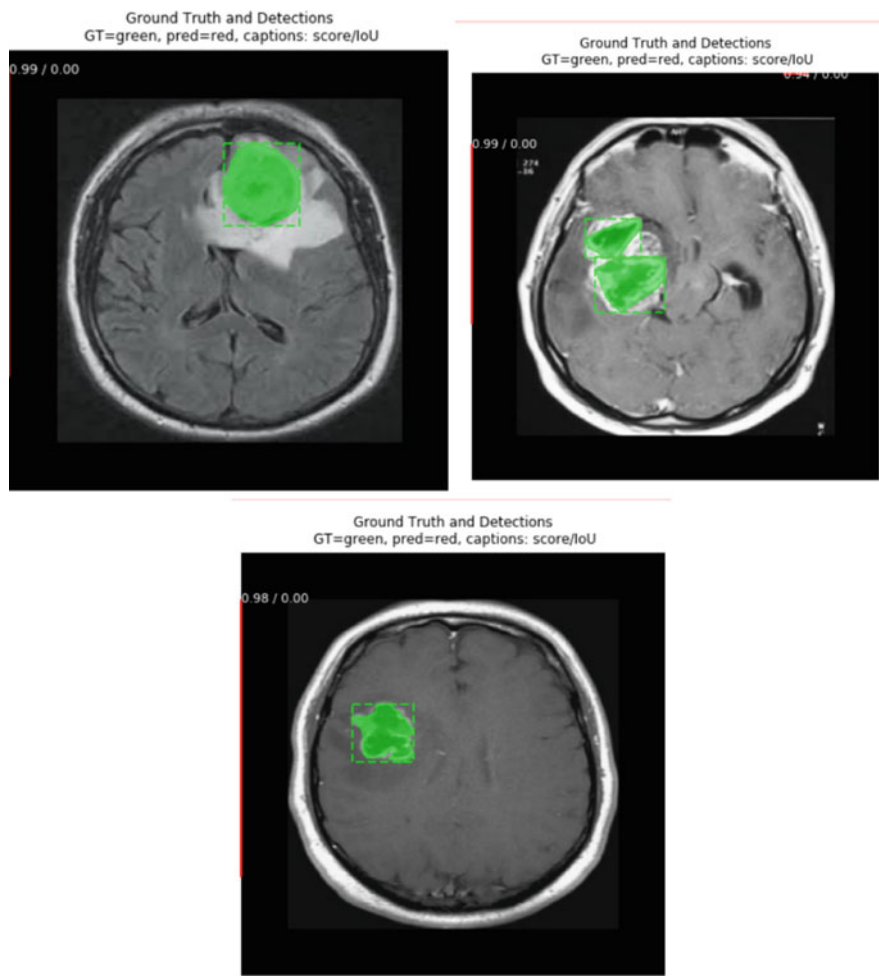


Fig. 9 Segmented brain tumor region by Mask R-CNN

References

1. VGG 16 - convolutional neural network model. <https://neurohive.io/en/popular-networks/vgg16/>. Accessed: 2021-1-8
2. Amin, J., Sharif, M., Gul, N., Yasmin, M., & Shad, S. A. (2020). Brain tumor classification based on DWT fusion of MRI sequences using convolutional neural network. *Pattern Recognition Letters*, 129, 115–122. <https://doi.org/10.1016/j.patrec.2019.11.016>.
3. Badža, M.M., Barjaktarović, M.C.: Classification of brain tumors from mri images using a convolutional neural network. *Applied Sciences (Switzerland)* **10**(6) (2020). <https://doi.org/10.3390/app10061999>.
4. Charron, O., Lallement, A., Jarnet, D., Noblet, V., Clavier, J. B., & Meyer, P. (2018). Automatic detection and segmentation of brain metastases on multimodal MR images with a deep convolutional neural network. *Computers in Biology and Medicine*, 95,. <https://doi.org/10.1016/j.combiomed.2018.02.004>.
5. Cheng, J.: brain tumor dataset (2015). <https://doi.org/10.6084/m9.figshare.1512427.v1>. https://figshare.com/articles/dataset/brain_tumor_dataset/1512427.
6. Daimary, D., Bora, M. B., Amitab, K., & Kandar, D. (2020). Brain Tumor Segmentation from MRI Images using Hybrid Convolutional Neural Networks. *Procedia Computer Science*, 167, 2419–2428. <https://doi.org/10.1016/j.procs.2020.03.295>.
7. Das, S., Aranya, O.F.R., Labiba, N.N.: Brain Tumor Classification Using Convolutional Neural Network. In: 1st International Conference on Advances in Science, Engineering and Robotics Technology 2019, ICASERT 2019 (2019). <https://doi.org/10.1109/ICASERT.2019.8934603>
8. Kaldera, H.N., Gunasekara, S.R., Dissanayake, M.B.: Brain tumor Classification and Segmentation using Faster R-CNN. In: 2019 Advances in Science and Engineering Technology International Conferences, ASET 2019 (2019). <https://doi.org/10.1109/ICASERT.2019.8714263>
9. Khalili, N., Lessmann, N., Turk, E., Claessens, N., de Heus, R., Kolk, T., et al. (2019). Automatic brain tissue segmentation in fetal MRI using convolutional neural networks. *Magnetic Resonance Imaging*, 64,. <https://doi.org/10.1016/j.mri.2019.05.020>.
10. Khan, H. A., Jue, W., Mushtaq, M., & Mushtaq, M. U. (2020). Brain tumor classification in MRI image using convolutional neural network. *Mathematical Biosciences and Engineering*, 17(5), 6203–6216. <https://doi.org/10.3934/MBE.2020328>.
11. Kumar, A., Manikandan, R., & Rahim, R. (2020). A Study on Brain Tumor Detection and Segmentation Using Deep Learning Techniques. *Journal of Computational and Theoretical Nanoscience*, 17(4), 1925–1930. <https://doi.org/10.1166/jctn.2020.8468>.
12. Kumar, S., Negi, A., Singh, J.N., Gaurav, A.: Brain Tumor Segmentation and Classification Using MRI Images via Fully Convolution Neural Networks. In: Proceedings - IEEE 2018 International Conference on Advances in Computing, Communication Control and Networking, ICACCCN 2018, pp. 1178–1181 (2018). <https://doi.org/10.1109/ICACCCN.2018.8748614>
13. Moeskops, P., Viergever, M. A., Mendrik, A. M., De Vries, L. S., Benders, M. J., & Isgum, I. (2016). Automatic Segmentation of MR Brain Images with a Convolutional Neural Network. *IEEE Transactions on Medical Imaging*, 35(5), 1252–1261. <https://doi.org/10.1109/TMI.2016.2548501>.
14. Parihar, A. S. (2018). A study on brain tumor segmentation using convolution neural network. *Proceedings of the International Conference on Inventive Computing and Informatics, ICICI*, 2017, 198–201. <https://doi.org/10.1109/ICICI.2017.8365336>.
15. Pei, L., Vidyaratne, L., Hsu, W.W., Rahman, M.M., Iftekhharuddin, K.M.: Brain tumor classification using 3D convolutional neural network. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 11993 LNCS, pp. 335–342 (2020). https://doi.org/10.1007/978-3-030-46643-5_33
16. Pereira, S., Pinto, A., Alves, V., & Silva, C. A. (2016). Brain Tumor Segmentation Using Convolutional Neural Networks in MRI Images. *IEEE Transactions on Medical Imaging*, 35(5), 1240–1251. <https://doi.org/10.1109/TMI.2016.2538465>.

17. Rajini, N. H. (2019). Brain tumor image classification and grading using convolutional neural network and particle swarm optimization algorithm. *International Journal of Engineering and Advanced Technology*, 8(3), 42–48.
18. Sankari, A., Vigneshwari, S.: Automatic tumor segmentation using convolutional neural networks. In: ICONSTEM 2017 - Proceedings: 3rd IEEE International Conference on Science Technology, Engineering and Management, vol. 2018-Janua, pp. 268–272 (2017). <https://doi.org/10.1109/ICONSTEM.2017.8261291>
19. Siar, M., Teshnehlal, M.: Brain tumor detection using deep neural network and machine learning algorithm. In: 2019 9th International Conference on Computer and Knowledge Engineering, ICCKE 2019, pp. 363–368 (2019). <https://doi.org/10.1109/ICCKE48569.2019.8964846>
20. Ucuzal, H., Yasar, S., Colak, C.: Classification of brain tumor types by deep learning with convolutional neural network on magnetic resonance images using a developed web-based interface. In: 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies, ISMSIT 2019 - Proceedings (2019). <https://doi.org/10.1109/ISMSIT.2019.8932761>
21. Wang, G., Li, W., Ourselin, S., Vercauteren, T.: Automatic brain tumor segmentation using convolutional neural networks with test-time augmentation. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 11384 LNCS, pp. 61–72 (2019). https://doi.org/10.1007/978-3-030-11726-9_6