

# Brain Tumor Detection from Brain MRI Images Using Deep Learning Algorithms

Janhavi Pawar

*Department of Computing and Mathematics.  
Manchester Metropolitan University.  
Manchester, United Kingdom.  
[22511910@stu.mmu.ac.uk](mailto:22511910@stu.mmu.ac.uk)*

## ***Abstract:***

A brain tumor is a condition brought on by the growth of abnormal brain cells. Brain tumors are rare and come in a variety of types, it is challenging to forecast the survival rate of a patient who is prone to these. according to cancer study conducted in the United Kingdom, around 15 out of every 100 persons with brain cancer will be able to survive for ten or more years after being diagnosed. One of the most challenging issues in medical image processing is the detection of brain tumors. Because brain tumors can have a variety of shapes and textures, there is a lot of variability in the images, making the identification process challenging. Different types of cells give rise to brain tumors, and these cells can provide clues about the form, severity, and rarity of the tumor. Tumors can appear in a variety of places, and the location of a tumor may provide information about the type of cells that are responsible for it, aiding in a more accurate diagnosis. The difficulty of detecting brain tumors can be made worse by issues with illumination that plague practically all digital photographs. Deep learning algorithms are used to diagnose brain tumors using magnetic resonance imaging data as artificial intelligence advances. Strong magnetic fields and radio waves are used in the Magnetic Resonance Imaging (MRI) scanning technique to provide precise images of the interior body.

***Keywords:*** Brain Tumor, MRI images, Sequential Model, ResNet50 Model, Deep Learning.

## I. INTRODUCTION

A brain tumor is a condition caused by the abnormal development of brain tissue. In a normal state, our bodies make new cells that replace the damaged and ageing ones in a controlled way. However, tumor cells continue to grow uncontrollably in brain tumors. Using Magnetic Resonance Imaging [MRI] scanning, the presence of tumor is examined. And the MRI scanning is always diagnosed by the physician and depending on the results of diagnosis the later treatments should be started. So, this process is bit

time consuming and to overcome this, using Deep Learning models, a system is designed that can classify if the person has brain tumor or not. The presented approach detects the tumor based on the MRI images and helps physicians to make decisions and start treatments earlier than before.

For this system, Sequential and ResNet models are used. The approach includes augmenting the dataset, preprocessing of data and comparative analysis based on the models' results. The goal of the study is to develop a system that uses deep learning to analyze brain MRI images. The MRI pictures collection is used to accomplish the goal since it includes MRI images of people with and without the condition.

## ***Aim:***

The ambition in this project as explained above is to develop an efficient system that will help in detecting brain tumors from the MRI scans of the patients with the help of CNN, ResNet.

The other portions of this study are organized as follows: Section II offers an analytical description of the dataset used for the research along with a full justification of the dataset selection. The three Deep Learning models are introduced in Section III along with the optimization of their parameters, which is followed by a review of previous research on a similar topic. In Section IV, the experiment's design is covered. The comparative analysis of the experiment's outcomes is presented in Section V, and Section VI concludes with conclusions and recommendations for the future.

## ***Objective:***

The dataset used contains MRI scanned images of total 2496, out of which 1500 are tumorous and 996 of them are non-tumorous. The presented work focuses on developing a detection model that detects the tumor in the MRI scanned images. The detection model in general can be given as:

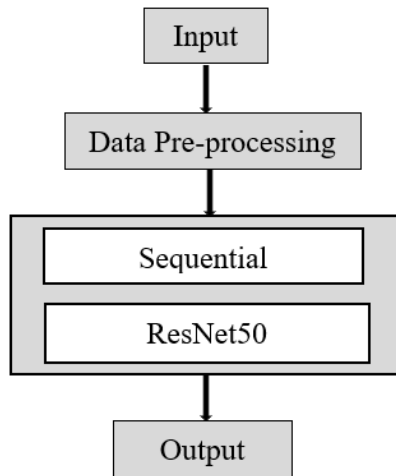


Fig.1 Flowchart of Brain Tumor Detection System

- A. Input:  
Brain MRI images are considered as input for this system.
- B. Data Pre-processing:  
The data is transformed into raw data for interpretation. Pre-processing includes importing packages, augmentation of data, converting images to grayscale, resizing images, cropping them using extreme points, splitting the dataset etc.
- C. Algorithms used in this system:
- 1) The Sequential Model
  - 2) Residual Network (ResNet)
- D. Output:  
Deciding which model would be the best fit to predict the tumor based on provided MRI images.

## II. RELATED WORK

A solution to develop brain cancer identification and confinement was put out by Abd-Ellah et al. They worked on the image's quality during preprocessing. Moreover, they modified the existing AlexNet neural layer to reduce complexity and accelerate model preparation. The setup provides 99.55% discovery accuracy using CNN, 92.95% using (VGG-16), and 95.15% using VGG-19. Furthermore, the restriction of a 0.87 dice score was met. The precision of this arrangement was 99.55%, which is very high when compared to 66.96% in earlier compositions [1].

The CAD approach is suggested by Devkota B et. al. and helps to detect brain tumors. In their investigation, the authors used mathematical morphological reconstruction. They have examined numerous preprocessed brain images in order to identify the locations

of the tumors. Following the completion of the preprocessing, the segregated images are classified to determine if the tumour is benign or malignant [2].

"Brain tumor segmentation based on a new threshold approach," UmitIlhan & AhmetIlhan [3]

- The median filter.
- Segmentation based on thresholds.
- The limited number of parameters for tumour detection.

According to Gao et al.'s research on early Alzheimer's disease detection, this disease impairs mental function in the brain. Two CNN models—2D and 3D—have been used; these models were trained using 2D and 3D Computerized Tomography images, and the outcome is determined by combining the output of these two models [4].

"Brain Tumor Detection from MRI Images Using Deep Learning Techniques", the article written by P Gokila Brindha et al. and published in the IOP Conference Series: Materials Science and Engineering in 2021. The study most likely describes the experimental setup and methodology utilized to create the deep learning model for brain tumor diagnosis. The precise deep learning architecture or architectures used, such as convolutional neural networks (CNNs) or more sophisticated models like ResNet or U-Net, may be described in detail. The deep learning model was probably created and tested using a collection of MRI scans that included both tumor and non-tumor cases [5].

Deep Learning models that are used in this project:

1. Sequential Model
2. Residual Network Model

Sequential Model:

A neural network architecture called the sequential model was created to recognize and categorize brain tumors from MRI data. Starting with convolutional layers that take the input images' spatial properties and patterns and extract them, it proceeds with a linear stack of layers. Pooling layers, which decrease the spatial dimensions and extract the most crucial characteristics, are added after these convolutional layers. In order to prepare the data for fully connected layers, a flattening layer transforms the multi-dimensional feature maps into a one-dimensional vector. High-level feature extraction and categorization are carried out by these dense layers. The predictions for brain tumor identification are produced by the output layer, the final layer. The summary of the model, which includes an optimizer and a loss function, gives a succinct summary of the architecture.

The sequential model is a good and dependable option for brain tumor detection tasks due to its simplicity, adaptability, effective training, and shown performance. To accurately identify brain tumors from MRI scans, it integrates well-established techniques for feature

extraction and classification. This enables researchers to take advantage of the potential of deep learning.

### 1) Residual Network (ResNet50):

A deep learning architecture called ResNet (Residual Neural Network) has demonstrated outstanding performance in several computer vision tasks, such as segmentation, object detection, and image categorization. The idea of residual connections is introduced, allowing the network to learn residual mappings rather than the desired underlying mapping directly. In deep networks, adding more layers might degrade the network's performance, this helps to solve the degradation problem.

The detection of brain tumors from MRI (Magnetic Resonance Imaging) pictures has been a common application for ResNet in the realm of medical imaging. The architecture is well-suited for analyzing elaborate and detailed medical images since it can effectively handle deep networks and capture intricate features.

Researchers have used pre-trained ResNet models, such as ResNet50 or ResNet101, that have been trained on extensive picture datasets like ImageNet, to detect brain tumors. These pre-trained models act as feature extractors, enabling the application of transfer learning to the goal of detecting brain tumors. The Residual Blocks idea was created by this design to address the issue of the vanishing/exploding gradient. We apply a method known as skip connections in this network. The skip connection bypasses some levels in between to link layer activations to subsequent layers. This creates a leftover block. These leftover blocks are stacked to create ResNets.

The strategy behind this network is to let the network fit the residual mapping rather than have layers learn the underlying mapping. Thus, let the network fit instead of using, say, the initial mapping of  $H(x)$ ,

$$F(x) := H(x) - x \text{ which gives}$$

$$H(x) := F(x) + x.$$

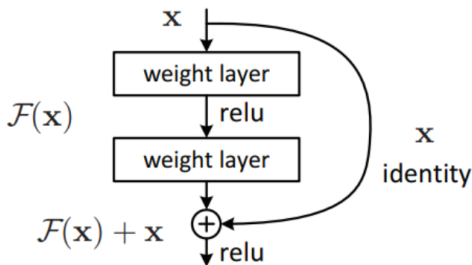


Fig 2. Skip (Short) Connection

This block is also called "Skip Connection" identity mapping, adds output from the previous layer to the following layer without using any parameters. The spatial resolution of an image is typically reduced by the convolutional procedure, even though  $x$  and  $F(x)$  have the

same dimensions. For example, a 3 3 convolution on a 32 32 image yields a 30 30 image. The inputs  $x$  and  $F(x)$  are combined as input to the layer ahead after the identity mapping is multiplied by a linear projection  $W$  to enhance the shortcut channels and match the residual. When both  $x$  and  $F(x)$  have different dimensions, equation III-C3 is employed.

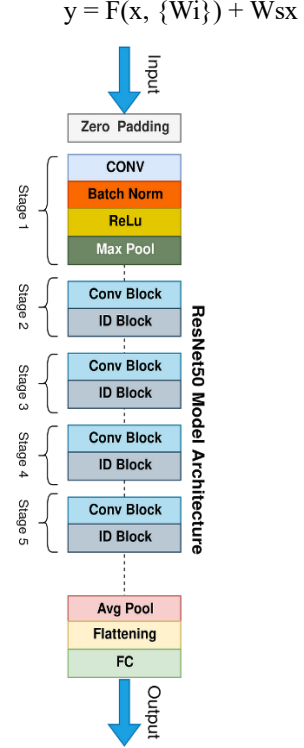


Fig. Illustration of ResNet50

### III. DATASET DESCRIPTION:

The dataset used is from Kaggle which is publicly accessible. The dataset contains 3000 images and is divided into two folders, 'yes' and 'no' representing the presence and absence of tumor respectively. Each picture shows a cross-sectional view of the brain that was taken with an MRI machine. The dataset also contains corresponding binary labels for each MRI image indicating whether a tumor is present. There are 1500 images in 'yes' and 'no' folders each. For making the process efficient I have deleted some pictures from 'no' folder. And for splitting, the dataset is split into 60% for train data, 20% for test data and 20% for validation data. Following are the images of 'no' and 'yes' folder i.e., normal and tumor respectively.

No Brain Tumor/ Normal:

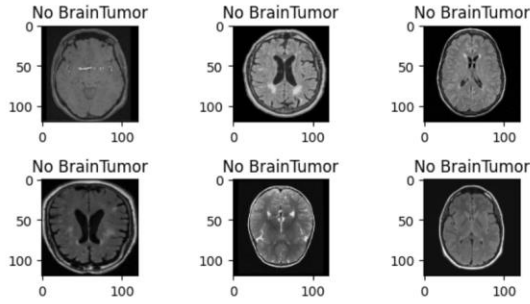


Fig 3. No Brain Tumor Images

Brain Tumor Present:

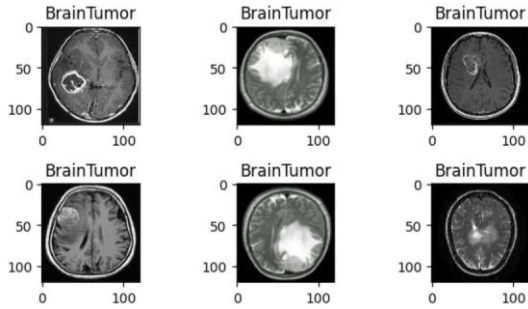


Fig 4. Brain Tumor Images

Distribution of Brain Tumor and Normal/ No Brain Tumor Images,

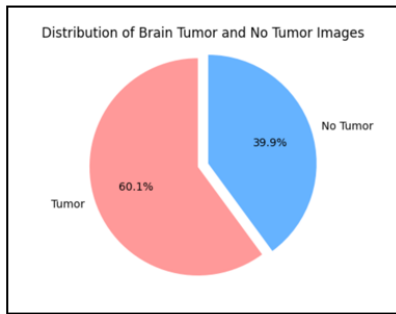


Fig.5 Distribution of Tumor and No Tumor in dataset

Data Pre-processing:

For data processing the training, validation, and test data folders are specified. The ImageDataGenerator from the TensorFlow Keras package is used to apply data augmentation and normalization methods to the training set of data. A variety of transformations, including rotating, shifting, shearing, zooming, and horizontal flipping, are applied to the training data to enhance it. Only normalization, not augmentation, is done to the validation and test data. The data generators are configured with a batch size of 32. For the training, validation, and test data folders, where the photographs are scaled to a target size of 244x244 pixels, data generators are made. Binary classification is enabled in the class mode. In conclusion,

the data is ready for the passive voice brain tumor detection model training.

#### IV. EXPERIMENTAL METHODOLOGY

In this study, a baseline Sequential model and ResNet50 are used. It was necessary to initially load the data from the various folders before training the models. The initial data analysis was performed on this dataset, which was then saved and accessed from cloud storage. The dataset was then combined with other data to create training, testing, and validation sets, and these sets were then expanded to train the models. According to the ratio parameter, which is set to (0.6, 0.2, 0.2), 60% of the data will be used for the training set, 20% for the validation set, and 20% for the test set. The random splitting method is consistent because the seed value is set to 1337.

##### 1. Sequential Model:

The performed model for this system, made up of dense layers, flatten layers, max pooling layers, convolutional (conv) layers, and a final output layer. Convolutional layers: Several convolutional layers are added with kernel sizes of (2, 2) and an increasing number of filters (32, 64, 128). Max pooling layer is added with a pool size of (2, 2) after each convolutional layer. By reducing the input's spatial dimensions, max pooling aids in the feature extraction process. Flatten layer is added to turn the 2D output from the convolutional layers into a 1D feature vector that can be fed into the dense layers. To accomplish high-level feature extraction, dense layers are added. A dropout layer with a dropout rate of 0.4 follows the first dense layer, which has 256 units, to lessen overfitting. Next, a dense layer with 128 units and a dropout layer are added. The final addition is a dense layer made up of 64 units. For binary classification tasks, the final dense layer's single unit with a sigmoid activation function is appropriate. The projected likelihood of the affirmative class is output (1 - the presence of a brain tumor). This was the approach for the Sequential model.

Epoch was set to 30 so that training dataset will be passed to the model 30 times. Then the test loss and test accuracy are calculated. Test loss is 0.357 and test accuracy is 0.85 i.e., 85%.

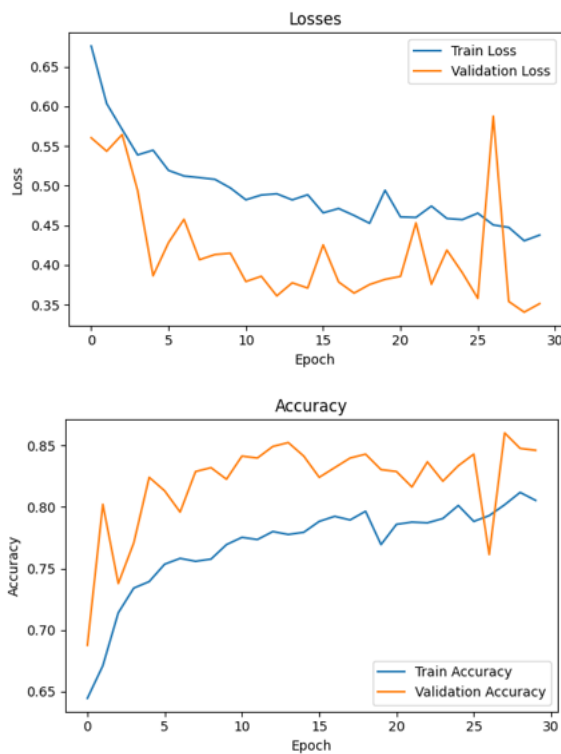


Fig 6. Performance of Sequential Model during Training

The model achieves a test loss of 0.357, indicating that, on average, the model's predictions differ from the true values in the test dataset by about 0.357 units. 85.1% of the cases in the test dataset were properly classified by the model, according to the test accuracy. The model's loss goes from a starting value of 0.676 to a finishing value of 0.431 during training. This results in a loss decrease of about 36.2%. The training accuracy increases from 64.4% to 81.2%, showing an increase of almost 16.8% in occurrences that are correctly identified. The model achieves validation losses for the validation set ranging from 0.340 to 0.588, with the lowest validation loss being 0.340. There is a 68.8% to an 86.0% range in the validation accuracy.

Accuracy increases as well, demonstrating that the model gains the ability to accurately categorize the training data. While the validation accuracy often rises, the validation loss fluctuates but follows a declining trend. These findings imply that the model generalizes effectively to new data. The model performs well on a different test dataset, achieving a low loss and excellent accuracy, further demonstrating its efficacy.

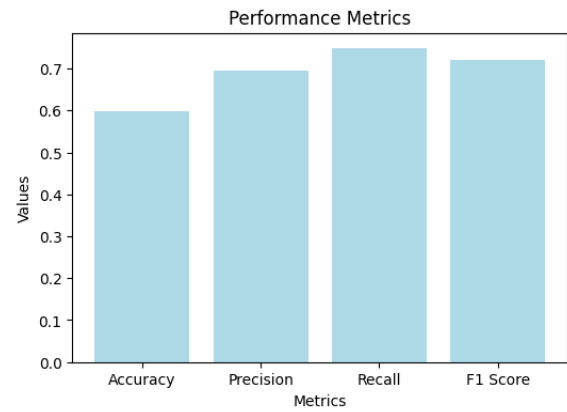


Fig 7. Performance metrics for Sequential Model

On the evaluation dataset, the supplied model performs at a mediocre level. It has a 59.8% accuracy rate, which means that 59.8% of the cases are classified properly. The model appears to minimize false positive predictions to a respectable level, as indicated by its precision of roughly 69.6%. A high percentage of the real positive events are identified by the model, according to the recall, which is roughly 74.8%. The model's effectiveness is further supported by the F1 score, which, at roughly 72.1%, offers a balanced assessment of the model's overall performance while taking precision and recall into account.

Confusion Matrix:

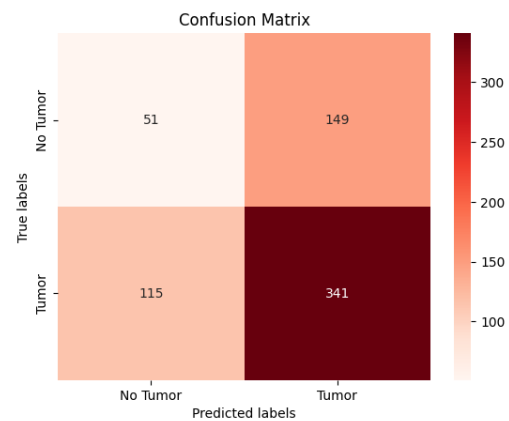


Fig 8. Confusion Matrix of the Predictions made by Sequential Model

ROC Curve:

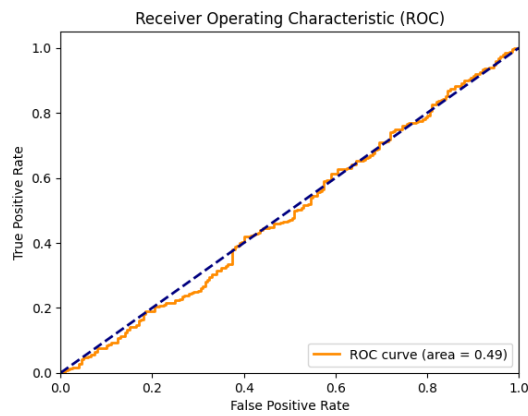


Fig 9. ROC curve of Sequential Model

The sequential model performs well in predicting labels for the test dataset, with a test accuracy of about 85.1% and a test loss of 0.3565. The model performs somewhat well in classification tasks, according to evaluation measures including precision, recall, and F1 score. The recall of 0.7478 shows that the model captures around 74.8% of the actual positive cases, while the precision of 0.6959 says that the model's positive predictions are accurate about 69.6% of the time. The precision and recall-balancing F1 score is 0.7209 points. The model struggles to distinguish between the negative and positive classes, according to the ROC curve analysis, demonstrating the need for additional optimisation to increase its discriminatory strength. Although the model generally shows good accuracy, there is still potential for improvement in terms of classification performance and the capacity to distinguish between positive and negative examples.

#### 1. ResNet50:

In this model, the top classification layer is not imported, and the pre-trained layers' weights are frozen to prevent changes during training. A global average pooling layer, a dense layer with 128 units and ReLU activation, and a dropout layer with a dropout rate of 0.3 are added as classification layers on top of the pre-trained model. There is an additional dense layer with 64 units and ReLU activation. Additionally included is the output layer with a single unit and sigmoid activation for binary classification. The Adam optimizer, binary cross-entropy loss function, and accuracy are then used to compile the model. The ResNet50 features are combined with extra layers to generate a model for a particular task of binary brain tumor detection. This is how the ResNet50 model was performed for the dataset.

Epoch was set to 30 so that training dataset will be passed to the model 30 times. Then the test loss and test accuracy are calculated. Test loss is 0.517 and test accuracy is 0.74 i.e., 74%.

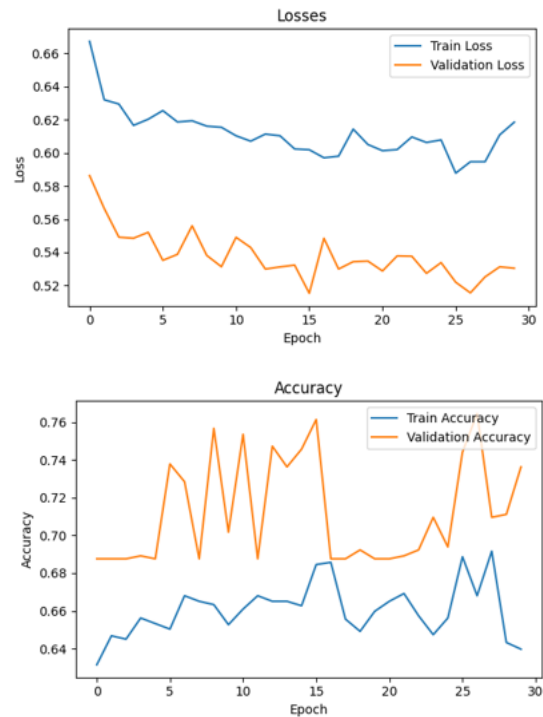


Fig 10. Fig Performance of ResNet during Training

#### Performance Matrix:

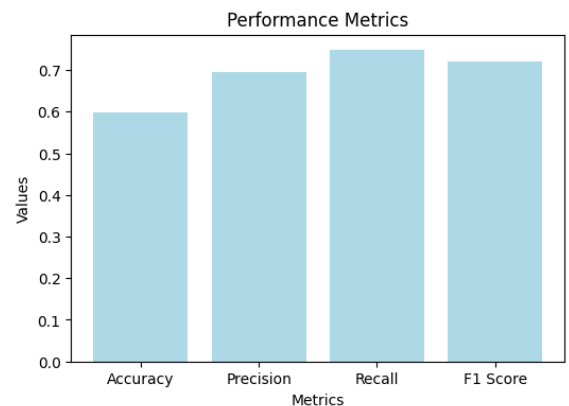


Fig 11. Performance metrics for ResNet50 model

The ResNet model performs admirably in predicting labels for the test dataset, with a test accuracy of roughly 74.5% and a test loss of 0.5174. The model performs somewhat well in classification tasks, according to evaluation measures including precision, recall, and F1 score. The recall of 0.7478 shows that the model captures around 74.8% of the actual positive cases, while the precision of 0.6959 says that the model's positive predictions are accurate about 69.6% of the time. The precision and recall-balancing F1 score is 0.7209 points.



Overall, the model's classification performance can be enhanced even though its accuracy is acceptable.

Confusion Matrix:

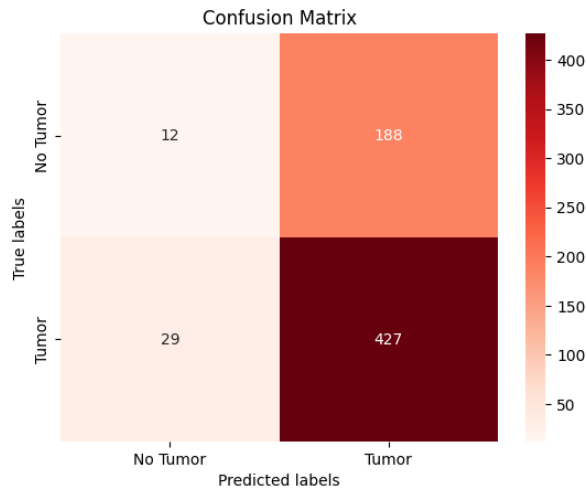


Fig 12. Confusion Matrix of the Predictions made ResNet50 Model

ROC Curve:

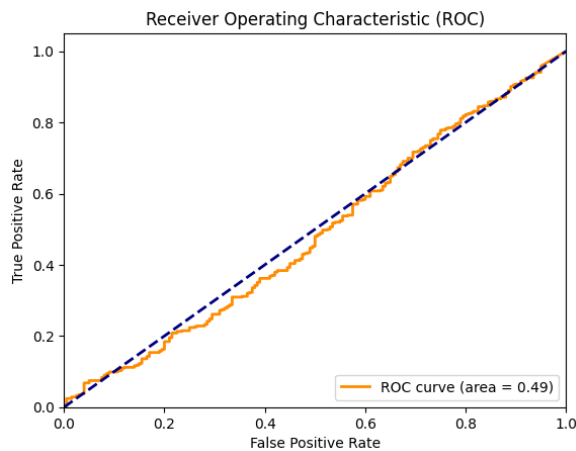


Fig 13. ROC Curve of ResNet50 Model

Overall, the ResNet model performs satisfactorily. It classifies 74.54% of the test data correctly, achieving a test accuracy of 74.54%. With a test loss of only 0.5174, the model appears to be capable of reducing training-related mistakes. The orange line's proximity to the blue line on the ROC curve, however, suggests that the model would have trouble discriminating positive from negative cases. As a result, despite the model's reasonable accuracy and loss performance, there may be room for improvement in terms of class distinction.

Model Comparison:

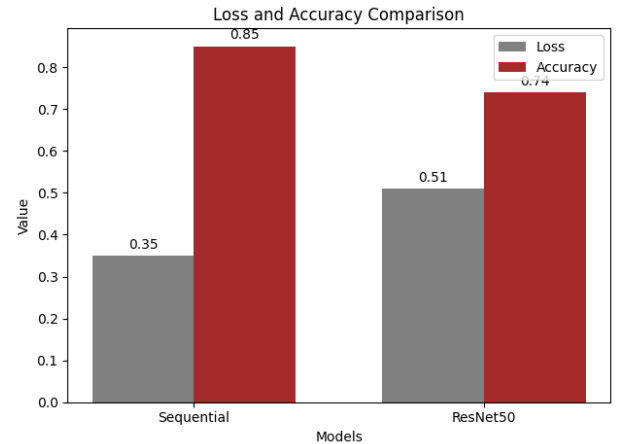


Fig 14. Comparison of Accuracy % of Scores of models

## V. EXPERIMENTAL RESULTS AND DISCUSSION:

Let's analyze the Sequential and ResNet models' performance in more depth:

### 1. Test Accuracy:

- Sequential Model: The Sequential model has an accuracy rate of 85.06%.
- ResNet Model: The ResNet model scores 74.54% on the test of test accuracy.

The percentage of accurately predicted labels on unseen data is the test accuracy statistic. Given that it has a greater test accuracy, the Sequential model outperforms the ResNet model in this scenario. This suggests that the Sequential model classified the test data more accurately.

### 2. F1 Score:

- Sequential Model: The Sequential model has an F1 score of 0.7209.
- ResNet Model: The ResNet model's F1 score is also 0.7209.

A model's accuracy is evaluated using the F1 score, which considers both precision and recall. It offers a fair assessment of a model's performance. The fact that both models in this instance have the same F1 score shows that their precision and recall are balanced similarly.

These measurements show that the Sequential model performs better in terms of test accuracy than the ResNet model. As a result, it can be said that the Sequential model performed better overall in this comparison.

## VI. CONCLUSION:

In conclusion, the performance of the Sequential Model and the ResNet Model in terms of accuracy, precision, recall, and F1 score is comparable. Indicating a similar balance of correctly classified cases, true positives, and true negatives, they obtain the same values for these measures. The Sequential Model, however, beats the ResNet Model in terms of test accuracy, with an accuracy of 85.06% as opposed to 74.54%. Thus, it may be concluded that the Sequential Model performs better at categorizing unknown data and providing precise predictions.

In this case, the Sequential Model is preferable than the ResNet Model in terms of overall performance. It achieves a smaller test loss and higher test accuracy, demonstrating an improved capacity for generalization and precise prediction on fresh data.

## VII. REFERENCES

- [1]. Siddaiah, N., Vignesh, C., Narendra, T., Pujitha, K. and Madhu, D. (2022). *Brain Tumor detection from MRI scans using VGG- 19 networks*. [online] IEEE Xplore. doi:<https://doi.org/10.1109/ICETET-SIP-2254415.2022.9791683>.
- [2]. Nanware, D., Taras, S., Navale, S. and Student (2022). *BRAIN TUMOR DETECTION AND CLASSIFICATION USING MRI IMAGES AND CNN ALGORITHM*. [online] *International Research Journal of Modernization in Engineering Technology and Science*, pp.2582–5208. Available at: [https://www.irjmets.com/uploadedfiles/paper/volume2/issue\\_10\\_october\\_2020/4164/1628083164.pdf](https://www.irjmets.com/uploadedfiles/paper/volume2/issue_10_october_2020/4164/1628083164.pdf) [Accessed 26 May 2023].
- [3]. Umit Ilhan & Ahmet Ilhan “Brain tumor segmentation based on a new threshold approach”, 9 th International Conference on theory and application of soft computing, computing with words and perception, ICSCCW-2017,24-25 August 2017. Budapest Hungary, Procedia computer science Vol.120(2017), pp. 580-587
- [4]. X.W. Gao, R. Hui, Z. Tian. (J2017). “Classification of CT brain images based on deep learning networks, Computer methods and programs in biomedicine”, pp.49-56, Jan2017. <https://doi.org/10.1016/j.cmpb.2016.10.007>.
- [5]. Gokila Brindha, P., Kavinraj, M., Manivasakam, P. and Prasanth, P. (2021). Brain tumor detection from MRI images using deep learning techniques. *IOP Conference Series: Materials Science and Engineering*, 1055(1), p.012115.doi: <https://doi.org/10.1088/1757-899x/1055/1/012115>.

## VIII. PROJECT PROTOTYPE

<https://colab.research.google.com/drive/1JZfbx95rVzkrCTJlglu5lcEQsPS30ftu?usp=sharing>