**Brain Tumor Detection From 2D MRI Images**

Imtiaz Ahmed Tamim Ahmed Sadat Tahmeed Azad

ID: 1706062 ID: 1706063 ID: 1706064

Sec: A2 Sec: A2 Sec: A2

Dept: EEE Dept: EEE Dept: EEE

Level: 3 Term:1 Level: 3 Term:1 Level: 3 Term:1

MD. Mahadi Hasan

ID: 1706065

Sec: A2

Dept: EEE

Level: 3 Term:1

***Abstract*—** **Brain tumor detection is one of the most crucial and arduous tasks in the field of biomedical image processing as manual classification can produce inaccurate predictions and diagnosis. Moreover, the task becomes very hard when there are a lot of images to be processed. Sometimes, it becomes quite hard to differentiate between normal tissues and brain tumors manually. The main objective of this project is to automate the process to increase accuracy and precision in identifying brain tumors from 2D images. In this project, we’ve proposed a method to identify the existence of brain tumor by using K Means clustering algorithm followed by traditional classifiers and Convolutional Neural network. We’ve used two datasets for this, the first one containing 253 gray MRI images that was used in both the Machine Learning part and Deep Learning part. The second data set contains more than 1000 brain tumors classified as different types of tumors and it was used in only the Deep Learning part. In our project we got best performance from transfer learning in both dataset 1 and 2 and gained a test accuracy of 94.12% with a sensitivity of \_\_ in dataset 1 and a test accuracy of \_\_ with a sensitivity of \_\_ in dataset 2. The main aim of this paper is accurately identifying brain tumors while having a good sensitivity and specificity to support the validity of detection from this model.**

***Keywords—*** ***Deep learning, Machine learning, K means clustering, Convolutional Neural Network, Transfer Learning*.**

# INTRODUCTION

Brain tumor detection from 2D MRI images is a crucial project as it’ll help reduce labour and increase accuracy in this task if it performs well. On the other hand, if we are not confident about it’s performance and can’t establish its validity then we’ll not be successful in achieving the objective of this project.

Brain tumor occurs when abnormal cells form within the brain. There are two types of brain tumors - Malignant and benign. Malignant tumors originate in the brain, grow faster and aggressively invade the surrounding tissues. It can spread to other parts of the brain and affect the central nervous system. Cancerous tumors can be divided into primary tumors, which start within the brain; secondary tumors which have spread from elsewhere and are known as brain metastasis tumors. On the other hand, a benign brain tumor is a mass of cells that grow relatively slow in the brain.

Hence early detection of brain tumors can play an indispensable role in improving the treatment possibilities and a higher gain of survival possibilities can be accomplished. That’s why it’s not very dependable on leaving this crucial task to manual identification as even setting the manual error aside sometimes the tumors can be very ill defined with soft tissue boundaries. So, it’s a very extensive task to obtain the accurate identification and segmentation of tumors from the human brain.

In this project, we are focusing on the identification of the existence of tumors rather than the segmentation part. We have proposed machine learning and deep learning based methods to reach the desired accuracy and validity so that we can help automate the process saving diagnosis time and misdiagnosis of brain tumor patients.

.

# DATA COLLECTION

For data collection we’ve used 2 datasets, both of them were collected from kaggle datasets.

The first one is the version of the BraTS(Brain Tumor Segmentation Challenge) dataset used as a benchmark in brain tumor related papers; which is in 2D format as we intended. The recent datasets are focusing towards 3D segmentation of brain tumors and so we didn’t use them as they don’t align with our objective.We used this dataset in both Machine Learning and Deep Learning part of our project. This dataset contains 155 tumor examples and 98 non-tumor examples, each of which is grayscale images.

The second dataset was also obtained from kaggle. It’s bigger than the previous one and so it's a more useful Deep learning part. We’ve used it only for the Deep Learning part of our project.

It contains 3370 images classified in 3 classes of tumors and 1 class of non-tumors. We used all of the tumor classes as tumor examples and the non-tumor class as non-tumor examples.

Both of the datasets are heavily biased towards positive examples meaning the  tumor dataset which was handled inside the implementation of our project for better results with image augmentation and imbalance handling.

| Dataset | Tumor data | Non-tumor data |
| --- | --- | --- |
| Dataset1 | 155 | 98 |
| Dataset2 | 2870 | 500 |

# **Proposed Methodology**

We have two main approaches to solve our problem, using Machine learning and deep learning. In the Machine Learning part we have preprocessed our dataset and segmented brain tumor using K-means clustering algorithm and then used multiple traditional classifiers on top of it. In the DL part we have used a convolutional neural network and a transfer learning model to differentiate tumor datasets from nontumor ones.

1. **Proposed methodology for classification using traditional ML classifiers:** Here at first, we’ve preprocessed images to have a suitable image dataset for extracting features to give as input in our classifiers. These preprocessed images give better results in terms of classification. After preprocessing we extracted features from these images and used them as input in our classifiers.  The whole process with explanation is illustrated in the following section



1. Skull Stripping:

Skull stripping is a very important step in preprocessing brain MRI images as most of the time skulls along with background don’t contain useful information about the problem at hand. In our project we’ve removed the skull in 2 steps:

* 1. Thresholding : For skull removal at first we’ve used adaptive thresholding which automatically calculates the thresholding value for each region defined by the filter size specified. It converts grayscale image to binary images which is crucial for performing different kinds of image processing operations

* 1. Connected component analysis: We’ve then used connected component analysis to get the desired brain or skull part separately as connected components. We’ve mainly got the skull extracted from here. Then we’ve recovered the brain part by subtracting the skull from the thresholded images.

2. Filtering and enhancement:

For better segmentation we need to maximize the MRI image quality and remove any external noise from there. Gaussian Blur filter was used to remove unwanted gaussian noises as they are most prevailing in medical image processing.

3. Segmentation using K means clustering algorithm:

K means clustering algorithm is used for segmenting the brain images, which allows the images to have multiple clusters, one of them containing the tumor part of the brain. This ensured better segmentation.

4. Morphological operations:

To extract the tumor we then used erosion to separate different parts of the segmented brain image and then dilation to connect the disconnected regions.

5. Tumor contouring:

Tumor extraction was done using thresholding again with connected component analysis. Then we have drawn a circle surrounding the tumor in the main image to get the contoured image which will be used as the output in our traditional classifiers.

6. Feature Extraction:

Two types of features were extracted for classification. Texture-based features such as- Dissimilarity, Homogeneity, Energy, Correlation,ASM, Entropy and statistical features as-Mean, Variance, Standard Deviation,Centroid, Kurtosis,skewness etc were extracted from the contoured images.

7. Traditional Classifiers:

We used six traditional classifiers which are - Naive Bayes, K-Nearest Neighbour, Support Vector Machine, Multilayer Perceptron, Ensemble trees(Random Forest), Logistic regression to get the tumor detection accuracy of our proposed model.

8. Evaluation stage:

After segmentation and feature extraction from the tumor we used the traditional classification techniques. Among them we’ve got the best results for Random Forest and the accuracy was 80.39%

1. **Proposed methodology for classification using Deep Learning techniques:**
2. **Baseline model using CNN:** Convolutional Neural Networks are the most used solution in terms of handling 2D image data and specially for medical imaging it has been playing a crucial role. It can consist of many layers like- convolutional layer, Pooling layer, dropout layer, input layer, fully connected layer etc. It’s mainly based on two processes: convolution using a trainable filter which has a pre-specified size and weights that are adjusted during the downsampling process in the training phase to have a high accuracy.

A five layer model is used as the baseline model which consists of six stages including a dropout layer. for solving this problem. Following is the proposed model methodology with brief explanation:









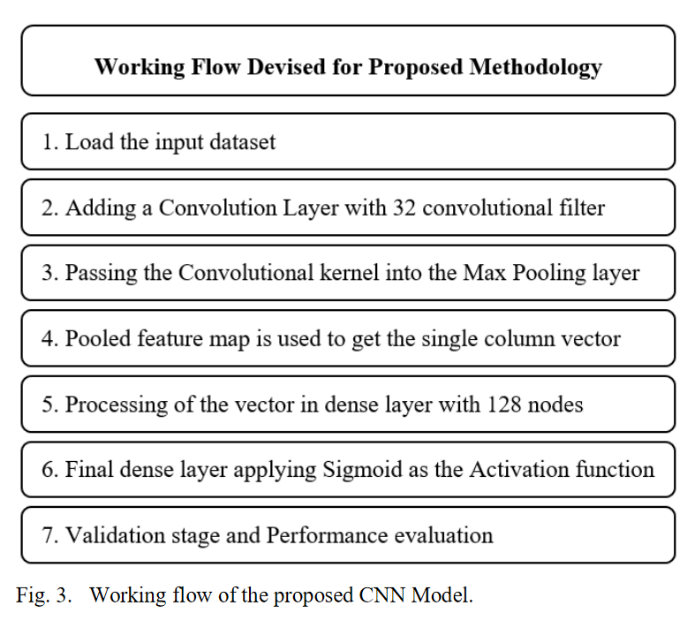






Using the convolutional layer as the first layer as the input layer an input shape of (224,224,,3) is used for all the MRI images. After accumulating all the images in the same aspect, we created a convolutional kernel that is convoluted with the input layer — administering with 32 convolutional filters of size 3\*3 each with the support of 3 channel tensors. ReLU is used as an activation function.

Then It’s followed by a 2x2 Max Pooling layer which has shrunk the image keeping all the relevant information. After the pooling layer, a pooled feature map is obtained. Flattening is one of the essential layers after the pooling because we’ve to transform the whole matrix representing the input images into a single column vector and it’s imperative for processing. It is then fed to the Neural Network for the processing. After it two fully connected layers are used which represent the dense layers. There are 128 hidden layers in Dense 1 layer and 1 layer in Dense 2 layer which also works as the classification layer with sigmoid activation function.  The working flow of our CNN model is shown below.



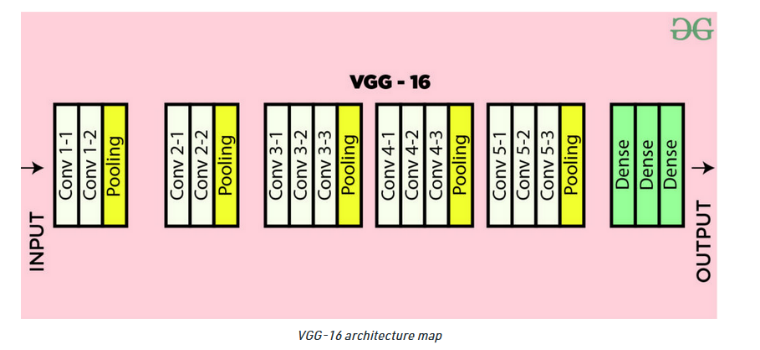
Using ADAM optimizer and Binary-Crossentropy as loss function we have compiled the model and found out the accuracy of the model.

All the hyperparameters are included in this are listed below :

| **Stage** | **Hyper-parameter** | **Value** | |
| --- | --- | --- | --- |
| Initialization | bias | Zeros | |
| Weights | glorot\_uniform | |
| Training | Learning rate | 0.001 | |
| beta\_1 | 0.9 | |
| beta\_2 | 0.999 | |
| epsilon | None | |
| decay | 0.0 | |
| amsgrad | False | |
| epoch | 100 | |
| **Stage** | **Hyper-parameter** | **Value** |
|  | Batch\_size | 8 |
| steps\_per\_epoch | 10 |

Here we get accuracy about 98.85% which is a very good result. We have altered architecture but the improvement wasn’t significant.

1. **Transfer Learning:** In this portion we have used different trained weights from transfer learning models like ResNet50,Efficient B0,MobilenetV2,VGG16 etc which are available in keras. The best performance was achieved by VGG 16 models for train test split being 80% and 20% for dataset 1 with an accuracy of 94.11% and for dataset-2 96.77%. As the datasets are very low and the model is very complex, we didn’t need to use augmentation for these. The transfer learning layers(VGG16) are illustrated in the following diagram:



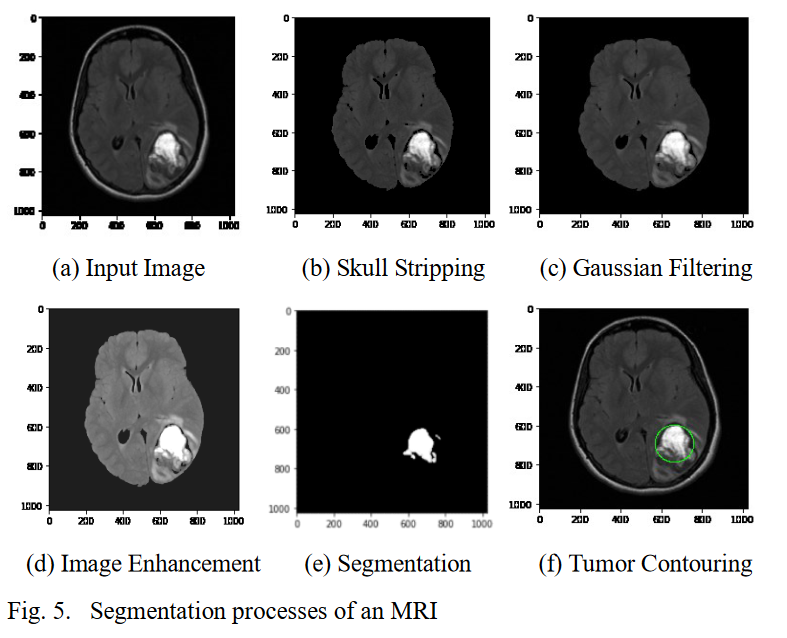
Here at first we’ve used a preprocessing layer that scaled the pixel values according to VGG 16 requirement and also used 224\*224\*3 images as standard values. Adam optimizer and Binary Crossentropy is used for compiling the model. The list of hyperparameters are given below:

| **Stage** | **Hyper-parameter** | **Value** | |
| --- | --- | --- | --- |
| Initialization | bias | Zeros | |
| Weights | glorot\_uniform | |
| Training | Learning rate | 0.0001 | |
| beta\_1 | 0.9 | |
| beta\_2 | 0.999 | |
| epsilon | None | |
| decay | 0.0 | |
| amsgrad | False | |
| epoch | 350 | |
| **Stage** | **Hyper-parameter** | **Value** |
|  | Batch\_size | 16 |
| steps\_per\_epoch | 13 |

# **Experimental results**

To increase the validity of our model we’ve included different models we’ve implemented in the process of finding the best solution and doing a comparative analysis among them. We’ve got the best accuracy in Machine Learning classifiers using Random Forest with accuracy 80% and in Deep Learning Transfer learning model with accuracy of 96.77% where we got 98.85% accuracy using our baseline model.

1. **Experimental Dataset:** For performance evaluation of our proposed model at first we used the benchmark dataset in the field of Brain Tumor Segmentation as our main dataset(Dataset-1); consisting of two classes-Tumor and non-tumor respectively having 155 and 98 images. After that for better performance in the Deep Learning part we’ve used another dataset containing 3370 tumor images and 500 non tumor images as our Dataset-2. For traditional classifiers we used a 80:20 split for our train and test dataset and for the Deep learning part we used 80:20 , 70:30 and also tried using a validation set with 80:10:10 to compare results among them.
2. **Segmentation using Image preprocessing Techniques:** Based on our methodology at first we have removed the skull as the skull does not contain any useful information. After that we’ve done filtering to enhance the images followed by segmentation using K -means algorithm. After that we used morphological operation to remove the disconnected regions from the segmented images and dilated afterwards to connect the important regions. After this we extracted the tumor so that later we can contour that region in our base images. This contoured image was used as input for Machine Learning Classifiers and Deep learning models afterwards as input.



1. **Classification using Machine Learning:** For classification using Machine Learning Classifiers we used two different types of approaches. Firstly, we used texture based features like - Contrast, Homogeneity, Dissimilarity, Directionality etc and statistical features like - Mean, Variance, standard deviation, skewness, energy, asm etc as our input to the traditional classifiers and then extracted performance from them. Table 2 depicts the values of some of the features used in classification. In the second approach we used contoured images as our input. After segmentation we have used these datasets in our ML classifiers namely K Nearest Neighbours, Naive Bayes, Logistic Regression, Random Forest, SVM. The best accuracy was achieved by the Random Forest algorithm. The values of confusion metrics along with the performance of the classifiers are included in table 3- The following factors evaluated performances-

**Accuracy =**

**Sensitivity(recall) =**

**Specificity =**

**Precision(PPV) =**

**Table II. Extracted Features from Segmented Tumor**

| **Image No** | **Image** | **Class** | **Mean** | **Standard Deviation** | **Variance** | **Skewness** | **Contrast** | **Homogeneity** | **Energy** | **ASM** | **Correlation** | **Dissimilarity** | **Directionality** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Y62.png | 1 | 37.97 | 41.86 | 1752.62 | 1.33 | 1181.18 | 0.29 | 0.10 | 0.01 | 0.63 | 16.36 | 287.28 |
| 1 | Y64.png | 1 | 63.39 | 62.66 | 3926.04 | 0.58 | 2007.8 | 0.33 | 0.28 | 0.08 | 0.73 | 23.33 | 331.88 |
| 2 | Y68.png | 1 | 44.65 | 40.41 | 1632.79 | 0.87 | 793.93 | 0.21 | 0.07 | 0.01 | 0.73 | 14.46 | 300.52 |
| 3 | Y71.png | 1 | 67.60 | 55.19 | 3046.11 | 0.75 | 1215.17 | 0.24 | 0.11 | 0.01 | 0.79 | 18.68 | 342.26 |
| 4 | Y73.png | 1 | 45.55 | 56.16 | 3154.18 | 1.53 | 1124.57 | 0.31 | 0.24 | 0.06 | 0.82 | 18.12 | 391.87 |

**TABLE III. CONFUSION METRICS’ OF THE CLASSIFIERS**

| **Classifiers** | **Accuracy** | **Recall** | **F1 Score** | **Specificity** | **Precision** |
| --- | --- | --- | --- | --- | --- |
| Naive Bayes Classifier | 64.71 | 0.75 | 0.702 | 0.63 | 0.66 |
| K Nearest Neighbours | 64.71 | 0.61 | 0.65 | 0.70 | 0.71 |
| Support Vector Machines | 72.55 | 0.82 | 0.7667 | 0.61 | 0.72 |
| Logistic Regression | 62.75 | 0.86 | 0.72 | 0.35 | 0.62 |
| Bagging | 80.39 | 0.96 | 0.84 | 0.61 | 0.75 |
| Decision tree | 72.55 | 0.93 | 0.7855 | 0.48 | 0.68 |
| Neural Network | 66.66 | 0.79 | 0.725 | 0.52 | 0.67 |
| Random Forest | 80.39 | 0.96 | 0.842 | 0.61 | 0.75 |

From table 3 we see that Random Forest gives the most prominent result in terms of accuracy it is 80.39%. But in terms of specificity KNN algorithm performs better. But the discrepancy is very little and it performed better for other metrics like recall,f1 score.

1. **Classification with baseline CNN model:** The five layer methodology gave very good results in terms of accuracy. Convolution,Max Pooling,Flatten,2 Dense layers are the layers of the proposed model. Data Augmentation is not needed as we got good results with such a simple network. In dataset 1, we have accuracy of  98.79% without using data augmentation with 70:30 train test split. We have accomplished 98.70% accuracy using a 80:20 train test split. Later we also used a validation dataset with 80:10:10 split and got 96.63% accuracy. In dataset 2, we used 80:20 split in train and test dataset and got \_\_% accuracy and using 80:10:10 train validation and test split we got \_\_% accuracy.

For the first basic model we set initial weight and oversampled to handle the data imbalance but the output didn’t improve. Table 4 represents our performance based on the CNN model.

The result is very good for a five layer CNN model. We have used many different types of combination but the outputs didn’t differ much and the network was already overfitting in this small network. Furthermore we used a 0.2 as the dropout but that reduced the performance a bit. As a result this model performs best without dropout.

for dataset 1:

without validation set:

| **No** | **Training Image** | **Testing Image** | **Splitting Ratio** | **Accuracy (%)** | **Precision(%)** | **Recall(%)** | **F1\_score(%)** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 176 | 77 | 70 : 30 | 98.79 | 98.98 | 99.06 | 99.02 |
| 2 | 202 | 51 | 80 : 20 | 98.70 | 98.93 | 98.96 | 98.94 |

with validation set:

| **No** | **Training Image** | **Validation Image** | **Testing Image** | **Splitting Ratio** | **Accuracy (%)** | **Precision(%)** | **Recall(%)** | **F1\_score** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 189 | 43 | 21 | 75:15:10 | 95.75 | 95.14 | 97.73 | 96.12 |
| 2 | 202 | 25 | 26 | 80 : 10:10 | 96.63 | 97.39 | 97.08 | 97.23 |

We use a callback function so that the training doesn't overfit and we can restore the best weights.

for dataset 2:

without validation set:

| **No** | **Training Image** | **Testing Image** | **Splitting Ratio** | **Accuracy (%)** | **Precision(%)** | **Recall(%)** | **F1\_score** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 176 | 77 | 70 : 30 | 96.23 | 97.45 | 98.31 | 97.88 |
| 2 | 2612 | 653 | 80 : 20 | 89.17 | 92.17 | 95.93 | 94.01 |

1. **Transfer Learning Model:** We used different transfer learning models like EfficientNet B0, Resnet50, MobilenetV2, VGG16,VGG19 etc and compared their performances. We got the best result from the VGG16 model for Dataset 1 at 80:20 train test split and \_ for Dataset 2 at 80:20 train test split. Their performance comparison is shown for dataset 1 in table 5 and dataset 2 in table 6

**table 5: accuracy for dataset 1 (80:20 split)**

| Classifiers | Accuracy |
| --- | --- |
| Resnet50 | 94.12 |
| Efficient B0 | ````96.08 |
| VGG16 | 96.08 |
| VGG19 | 96.08 |
| MobilenetV2 | 94.12 |

**table 6: accuracy for dataset 2 (80:20 split)**

| Classifiers | Accuracy |
| --- | --- |
| Resnet50 | 93.12 |
| Efficient B0 | ````94.08 |
| VGG16 | 94.08 |
| MobilenetV2 | 91.12 |

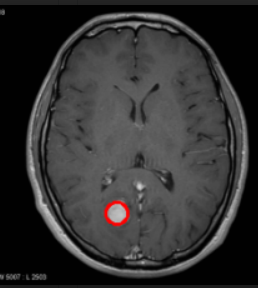
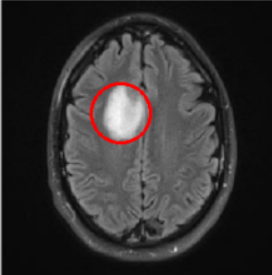
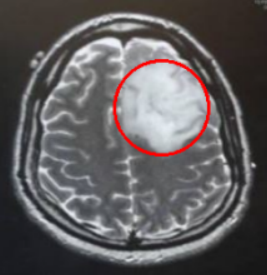
1. **Performance Comparison:** Finally we carried out a comparison between our proposed methodologies, which are classification of brain tumors between 2 classes (tumor vs non-tumor) using traditional ML classifier, CNN model and Transfer learning. Our proposed methodology worked well in CNN and Transfer Learning in both datasets, but we didn’t achieve good results in our ML dataset. We suspect the reason being our inability to preprocess the dataset 1 well enough as we faced difficulties extracting brain tumor perfectly. For that reason we achieved only 80% test accuracy in Random Forest which is the highest among the ML classifiers. But overall we reached our expected accuracy level through Deep Learning implementation. A comparison among the best results in the proposed 3 alternatives for dataset 1 are illustrated in the following table.

| **Method** | **Accuracy** |
| --- | --- |
| Machine Learning | 80.39(Random Forest) |
| Deep learning(CNN architecture) | 98.79%(70:30 split) |
| Transfer Learning | 96.08%(VGG 16) |

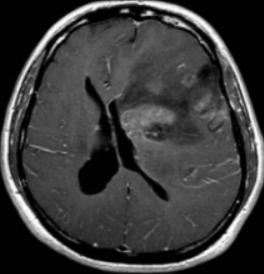
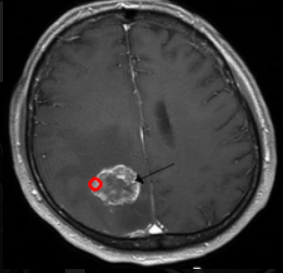
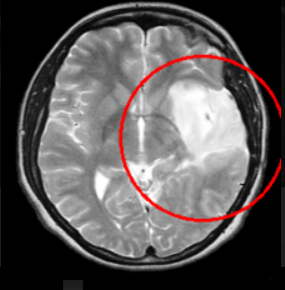
1. **Challenges faced:**

In the Machine learning part, we mainly faced problems with the preprocessing of images. We couldn’t get a perfectly segmented tumor and as a result our segmentation was not perfect. We couldn’t also automate the preprocessing part as different parameters like cluster number , width,height ,area of connected components varied from image to image. Also while segmenting many times For these reasons the accuracy of our classifiers is not good. In some of the images we couldn’t identify the tumor by using image processing. We used their main picture as our input later.

**Good cases:**

**  **

**Error cases:**

no contour found contour very small than the tumor contour is very big than the tumor.

In the deep learning part, the main problem was overfitting and imbalance the dataset. Although we have got a very good result there is class imbalance, which we can see in the case of dataset 2, as it has lower accuracy than dataset 1. So for larger dataset to have more accuracy we need to handle dataset imbalance and overfitting using augmentation.In the transfer learning part the same problems remained, overfitting and imbalanced dataset. Also we noticed there were some images that were very hard to identify tumors even manually. Though the effect in accuracy is mainly prevailing in the Machine Learning part. They were challenging to identify. The class ratio of the dataset is shown below.

| Dataset | tumor : non-tumor ratio |
| --- | --- |
| Dataset 1 | 1.58 |
| Dataset 2 | 7.265 |

1. **Conclusion and Future work:** In medical image analysis, brain tumor classification is a crucial task and a lot of the time diagnosis of cancer patients depends on correct diagnosis of brain tumor. For our project we have used 2D MRI images of brain tumors and implemented different solutions for correctly diagnosing tumor datasets. For better results we segmented and contoured our dataset 1 followed by traditional classifiers in which we couldn’t get a promising result. But our CNN (with a split ratio 80:20) and transfer learning model(VGG 16) performed pretty well in both datasets. We have used dataset 2 for identifying how our deep learning model responded to additional data.

In the future, we would like to experiment with 3D segmentation of brain MRI and try to correctly segment brain tumors and lesions. For that we may use U net architecture. We have also planned to implement an Instance segmentation and semantic segmentation of 3D brain MRI in future to correctly identify cancerous tissues and other lesions beside brain tumors to perform better diagnosis for disease and abnormalities from Brain MRI for ensuring better result in terms of early preventions of brain related diseases. If we can implement a more accurate model for that with better precision and specificity it can be a great step towards early detection and reduction of brain tumor related diseases.

# REFERENCES

1. Tonmoy Hossain, Fairuz Shadmani Shishir, Mohsena Asraf, MD Abdullah Al Nasim, Faisal Muhammad Shah(2019). “Brain Tumor Detection Using Convolution Neural Network”. 1st International Conference on Advance in Science, Engineering and Robotics Technology(ICASERT2019).
2. Kasban, Hany & El-bendary, Mohsen & Salama, Dina. (2015). “A Comparative Study of Medical Imaging Techniques”. International Journal of Information Science and Intelligent System. 4. 37-58.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
3. D. Surya Prabha and J. Satheesh Kumar, “Performance Evaluation of Image Segmentation using Objective Methods”, Indian Journal of Science and Technology, Vol 9(8), February 2016.
4. Brain Tumor: Statistics, [Cancer.Net Editorial Board,](https://www.cancer.net/about-us) 11/2017 (Accessed on 17th January 2019)
5. Kavitha Angamuthu Rajasekaran and Chellamuthu Chinna Gounder, Advanced Brain Tumour Segmentation from MRI Images, 2018.
6. [General Information About Adult Brain Tumors".](http://www.cancer.gov/cancertopics/pdq/treatment/adultbrain/Patient/page1/AllPages) *NCI*. 14 April 2014. [Archived f](https://web.archive.org/web/20140705225928/http:/www.cancer.gov/cancertopics/pdq/treatment/adultbrain/Patient/page1/AllPages)rom the original on 5 July 2014. Retrieved 8 June 2014. (Accessed on 11th January 2019)
7. M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.
8. B. Devkota, Abeer Alsadoon, P.W.C. Prasad, A. K. Singh, A. Elchouemi, “Image Segmentation for Early Stage Brain Tumor Detection using Mathematical Morphological Reconstruction,” 6th International Conference on Smart Computing and Communications, ICSCC 2017, 7-8 December 2017, Kurukshetra, India.
9. Song, Yantao & Ji, Zexuan & Sun, Quansen & Yuhui, Zheng. (2016). “A Novel Brain Tumor Segmentation from Multi-Modality MRI via A Level-Set-Based Model”. Journal of Signal Processing Systems. 87. 10.1007/s11265-016-1188-4.
10. Ehab F. Badran, Esraa Galal Mahmoud, Nadder Hamdy, “An Algorithm for Detecting Brain Tumors in MRI Images”, 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence, 2017.
11. Pei L, Reza SMS, Li W, Davatzikos C, Iftekharuddin KM. “Improved brain tumor segmentation by utilizing tumor growth model in longitudinal brain MRI”. *Proc SPIE Int Soc Opt Eng*. 2017.
12. Dina Aboul Dahab, Samy S. A. Ghoniemy, Gamal M. Selim, “Automated Brain Tumor Detection and Identification using Image Processing and Probabilistic Neural Network Techniques”, IJIPVC, Vol. 1, No. 2, pp. 1-8, 2012.
13. Mohd Fauzi Othman, Mohd Ariffanan and Mohd Basri, “Probabilistic Neural Network for Brain Tumor Classification”, 2nd International Conference on Intelligent Systems, Modelling and Simulation, 2011.
14. A. Rajendran, R. Dhanasekaran, “Fuzzy Clustering and Deformable Model for Tumor Segmentation on MRI Brain Image: A Combined Approach,” International Conference on Communication Technology and System Design2011.
15. Sobhaninia, Zahra & Rezaei, Safiyeh & Noroozi, Alireza & Ahmadi, Mehdi & Zarrabi, Hamidreza & Karimi, Nader & Emami, Ali & Samavi, Shadrokh. (2018). “Brain Tumor Segmentation Using Deep Learning by Type Specific Sorting of Images”.
16. Gupta, Gaurav and Vinay Singh. “Brain Tumor segmentation and classification using Fcm and support vector machine.” (2017).
17. Anam Mustaqeem, Ali Javed, Tehseen Fatima, “An Efficient Brain Tumor Detection Algorithm Using Watershed & Thresholding Based Segmentation”, I.J. Image, Graphics and Signal Processing, 2012, 10, 34-39.
18. Seetha, J & Selvakumar Raja, S. (2018). “Brain Tumor Classification Using Convolutional Neural Networks. Biomedical and Pharmacology Journal”. 11. 1457-1461. 10.13005/bpj/1511.
19. Mariam Saii, Zaid Kraitem, “Automatic Brain tumor detection in MRI using image processing techniques”, Biomedical Statistics and Informatics, Vol. 2, No. 2, pp. 73-76, 2017.