

# Brain Tumor Detection

<sup>1</sup>Student, Ananaya Nanda Naik, <sup>2</sup>Student, Archana Anant  
Shanbhag, <sup>3</sup>Student, Trupti Suresh Deshbhandari, <sup>4</sup>Student,  
Tushara Naik, <sup>5</sup>Professor, Mr. Anal B

*Canara Engineering College Mangalore, Karnataka*  
*Canara Engineering College Mangalore, Karnataka*  
*Canara Engineering College Mangalore, Karnataka*  
*Canara Engineering College Mangalore, Karnataka*  
*Dept. of ISE, Canara Engineering College Mangalore, Karnataka*

Submitted: 01-06-2022

Revised: 10-06-2022

Accepted: 15-06-2022

**ABSTRACT:** Brain tumor is considered as one of the aggressive diseases amongst youngsters and adults. Brain tumors make up 85 to 90 percent of all number one tumors of the central nervous system (CNS). Every year, around 11,700 people are diagnosed with brain tumors. The 5-12 months survival charge for humans with a cancerous brain or CNS tumor is about 34 percentage for men and 36 percent for ladies. Brain Tumours are classified as: Glioma Tumor, Meningioma Tumor, Pituitary Tumour, etc. Proper treatment planning and accurate diagnostics should be implemented to improve life expectancy of the patients. The best technique for detecting brain tumors is magnetic resonance imaging (MRI). Scanning produces a large amount of data images. These images are examined by a radiologist. Manual testing may be flawed due to the degree of complexity involved in the growth of the brain and its structures. The primary controller of the humanoid system is human brain. The abnormal growth and division of cells in the brain lead to a brain tumor, and the further growth of brain tumours leads to brain cancer. In the area of human health, Computer Vision plays a significant role, which reduces the human judgment that gives accurate results. CT scans, X-Ray, and MRI scans are the common imaging methods among magnetic resonance imaging (MRI) that are the most reliable and secure.

**KEYWORDS:** Benign, Malignant, Pituitary, MRI.

## I. INTRODUCTION

An Intracranial Neoplasm or brain tumor is a mass that is formed inside the brain by the tissues surrounding the brain or the skull and directly affects human life. Tumors are classified on

two bases: Whether they are cancerous and the origin. Benign tumors are called benign. These are easy to distinguish and have a slow growth rate. Cancerous tumors are called Malignant. These are very aggressive and can be life-threatening as they are hard to detect. These tumors grow unevenly in the brain and apply pressure around them. The effect of pressure causes various disorders in the brain that affect the body and symptoms of such diseases in humans include: dizziness, headache, fainting attacks, paralysis, etc. Unlike benign tumors, malignant tumors grow unevenly and damage surrounding tissues.

When it comes to detecting a tumor, doctors can opt for either an X-ray or an MRI. MRI is suitable when all other tests do not provide sufficient information. MRI scans use magnetic and radio wave characteristics to create accurate images. Neurosurgeons most commonly prescribe MRI's as it provides them with sufficient information to detect even the smallest abnormalities. However, MRI uses magnetic waves, so it is unsuitable for patients with pacemakers and metal implants. Now once we have the scanned image of the brain, it is important to accurately detect the tumor, its size, and its location. All this information is needed by the neurosurgeon to complete the diagnosis. Here, computer-based image processing is useful. With the use of different segmentation techniques and feature extraction methods, we can accurately detect the tumor with almost 91 percent accuracy.

If a low-grade brain tumor is left untreated, it will most certainly progress to a high-grade, aggressive brain tumor. Patients with grade II glioma should be monitored and observed every 6-12 months using magnetic resonance imaging (MRI) or computer tomography (CT). Brain tumors

are the growth of foreign cells in the brain or skull; a few are benign, whilst others are malignant. Tumors can originate from the mind tissue itself (number one) or most cancers from some other place inside the body may spread to the brain and its treatment depends on the location, size and type of tumor and the stage.

## II. IMPLEMENTATION

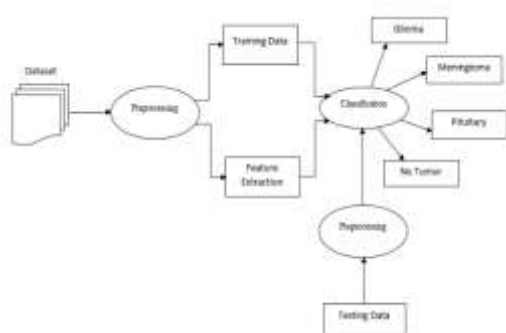


Fig i. Architectural Design

**Data:** Our project aims to classify MRIs into four classes which makes it a four-class problem. Data for our Neural Networks is in image format(.jpg). This data is collected from Kaggle. This data is unclean and has few incorrectly placed images, which were corrected, deleted or replaced. Also, a few images from other medical publications were added. The amount of data in the No-Tumor class was too low, hence few images of no-tumor were added from Google. To avoid distortion of the neural network, we need to get about the same number of images in each class.

**Pre-Processing:** The image data gathered from GitHub has three classes of the tumor while Kaggle data has the class of no-tumor. These images were split in the ratio of 7:3 for the Training and Testing phase respectively.

**Image Cropping:** The MRIs contain a black background around the central image of the brain. This black background provides no useful information about the tumor and would be waste if fed to neural networks. Hence cropping the images around the main contour would be useful. For this we use cv2.findContours() from the 'cv2' library.

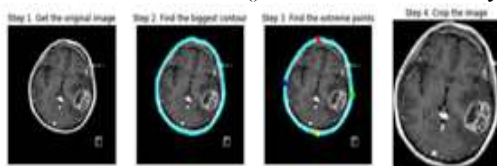
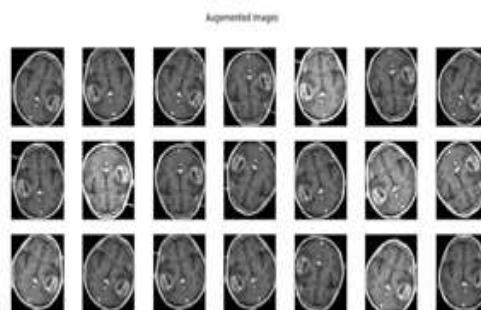


Fig.ii Steps in Image Cropping using cv2

Here we can see how the biggest contour is selected and marked. Next, we find the extreme points of the contour and crop the image on those endpoints. Thus, we have removed most of the unwanted background and some noise present in the original image. This process is performed for all the image in the dataset. However, note that sometimes cv2.findContours() may not be able to correctly recognize the correct contours and makes a mistake and wrongly crops the image. Such images should be removed by manual inspection before entering the 'augmentation' phase.

**Augmentation:** The amount of data gathered was very low and could cause the models to under-fit. Hence, Use of good techniques for data expansion to increase the amount of data. This technique relies on rotations, flips, change in exposure, etc to create similar images. Using this technique, we can increase the size of data by a high factor. The output image of the cropping stage is given as input to Image Data Generator which is a function in Keras, pre-processingimagelibrary. This function takes multiple arguments that decide how Augmentation takes place.



Fi

g.iii Augmentation

The

output of ImageDataGenerator with defined necessary arguments is multiple images which are then saved to a separate folder. The process of augmentation is applied to every image of the dataset hence increasing the size of the dataset. Thus, with such a huge amount of data, the chances of underfitting can be lowered.

**Pickling:** The data after all the above steps are still in separate image file formats stored in respective folders. But to feed the data to a Neural Network it must be converted into NumPy array. Thus, we convert these images into four NumPy arrays of X\_train, Y\_train, X\_test, Y\_test. Where X\_train, X\_test contain the images and Y\_train, Y\_test

contain the labels. The data is read using cv2.imread(path\_of\_image,cv2.IMREADCOLOR). Here cv2.IMREAD\_COLOR is important, as it would specify the 'dimension of our input later' of the neural network. Next we use np.reshape(-1,IMG\_SIZE,IMG\_SIZE,3) to reshape the data stored in X\_train, X\_test. The number 3 indicates this is an RGB image and not a greyscale image. The IMG\_SIZE is to specify the resolution of images; higher resolution images contain more detail but at cost of greater pickle size and lower IMG\_SIZE may lead to pixelated images but saves storage and makes the network light. So, it's important to find a sweet spot for resolution. (We selected this as 150) Once this processing, reshape and splitting of images and labels of all images in train and test set is done we save the files as pickle files. Saving these files as pickle helps us save all our progress in the pre-processing stage using 'pickle.dump()'. And we can later directly use these pickle files to be input to all our neural networks. Thus, it helps us save time by not pre-processing every time we want to train a network. The next time we just use 'pickle.load' to load all the data.

**Model Building:**

**Convolutional Neural Network:** Convolutional neural networks (CNNs) are one of the most frequently used variants of neural networks in the field of laptop vision. Its name comes from the type of hidden layer that makes it up. The hidden layers of CNN include convolutional layers, pooling layers, fully connected layers, and standardized layers. CNN's are very good at feature extraction, thus they can learn features of all four classes. CNN training is a task of learning filters. The pooling layer just looks at the block of input data and passes the maximum value. This reduces the size of the output and eliminates the need to learn additional parameters, so pooling layers are often used to resize the network and keep the system below computational limits. The activation function gives the function non-linearity.

Thus, we generate a few CNN models of dense layers of 0,1,2 with layer size of 32,64,128 and a number of convolution layers of 1,2,3. Using these parameters, the model is generated. Thus, we generate a total of 21 models. In which the model '3-conv-128-nodes-1- dense' has the highest accuracy of 91%.

Below is the code to implement a convolutional neural network.

```
dense1 = [0,1,2]
layers = [32,64,128]
conv1 = [1,2,3]
```

```
for dense in dense1:
    for layer in layers:
        for conv in conv1:
            NAME = "{}-conv-{}-nodes-{}-dense-{}".format(conv,layer,dense)
            print(NAME)
            model = Sequential()
            model.add(Conv2D(layer, (3, 3),
                input_shape=X_train.shape[1:]))
            model.add(Activation('relu'))
            model.add(MaxPooling2D(pool_size=(2, 2)))
            for l in range(conv-1):
                model.add(Conv2D(layer, (3, 3)))
                model.add(Activation('relu'))
            model.add(MaxPooling2D(pool_size=(2, 2)))
            model.add(Flatten())
            for _ in range(dense):
                model.add(Dense(layer))
                model.add(Activation('relu'))
            model.add(Dropout(0.25))
            model.add(Dense(4))
            model.add(Activation('softmax'))
            model.compile(loss='sparse_categorical_crossentropy',
                optimizer="adam", metrics=['accuracy'],)
            #Fit the model
            history = model.fit(X_train, Y_train,
                batch_size=32,
                epochs=20, validation_data=(X_test, Y_test),
                callbacks=[tensorboard, es])
```

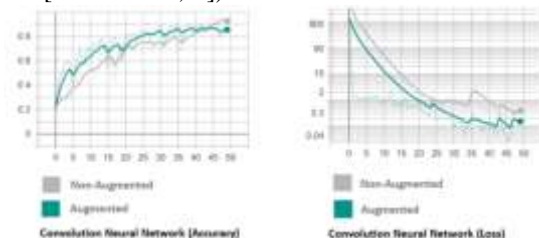


Fig. iv CNN accuracy, loss vs epoch

**Test Cases:** The test cases used to classify brain tumors

No	Input (Dataset)	Description	Expected Output	Actual Output	Status
1	Glioma tumor (MRI)	MR Image of glioma tumor classification	Glioma tumor	Glioma tumor	Passed
2	Meningioma tumor (MRI)	MR Image of Meningioma tumor classification	Meningioma tumor	Meningioma tumor	Passed
3	No tumor (MRI)	MR Image without any tumor classification	No tumor	No tumor	Passed
4	Pituitary tumor (MRI)	MR Image of Pituitary tumor classification	Pituitary tumor	Pituitary tumor	Passed

Table 1. Test cases for brain tumor classification

### III. RESULT

We have developed a website that detects brain tumors and classifies tumor types. The website includes a home page, detection page and implementation page. The proposed model had obtained an accuracy of 91% and yields promising results without much errors and less computational time.



Fig. v Brain tumor detection result

### IV. CONCLUSION

In the medical field, brain tumor classification is very important. We focused on creating a CNN classifier that distinguishes between three major tumor types (glioma, meningioma, pituitary), as well as no tumor. The proposed model begins with pre-processing of image data. Feature extraction aims to reduce the number of features in the dataset by creating new features from existing features. The proposed CNN model has been trained and tested with a sufficient number of medical images. The classification model is the final result of machine learning in a classifier. The model is trained using the classifier, so that the model, ultimately, classifies the different types of brain tumor.

### REFERENCES

- [1]. G Hemanth, M. Janardhan, L. Sujihelen. Design and implementing Brain tumor detection using Machine Learning approach
- [2]. Sneha Grampurohit, VenkammaShalavadi, Vaishnavi R Dhotargavi, MeghaKudari, Mrs. Soumya Jolad. Brain tumor detection using Deep Learning Model doi: 10.1109/INDISCON50162.2020.00037
- [3]. Avigyan Sinha, Aneesh R P, Nitha Mohan R, Malvika Suresh, Ashwin G Singerji, Abhinaya D. Brain tumor detection using Deep Learning. doi: 10.1109/icbsii51839.2021.9445185
- [4]. Aryan Sagar M ethil, Brain tumor detection using Deep Learning and Image processing. doi: 10.1109/icaais50930.2021.9395823.
- [5]. Sultan B. Fayyadh, Abdullahi A. Ibrahim, Brain Tumor Detection and Classification using CNN Algorithm and Deep Learning techniques.
- [6]. Subhashis Banerjee, Sushmita Mitra, Brain Tumor Detection and Classification from multi-channel MRI using Deep Learning and Transfer Learning.
- [7]. M. Aarthilakshmi, S. Meenakshi, A. Poorna Pushkala, V.Ramalakshmi, N.B Prakash (4 April 2020) Brain tumor detection using Machine Learning.
- [8]. P GokilaBrindha IOP Conf. Series: Material Science and Engineering, Brain tumor detection from MRI Images using Deep Learning techniques (). doi: 10.1088/1757-899X/1055/1/012115
- [9]. Suresha D, Jagadisha N, Shrisha H.S, Kaushik K.S, Detection of Brain Tumor using Image Processing.
- [10]. Zhesu Jia, Deyun Chen, Brain Tumor Identification and Classification of MRI images using Deep Learning techniques.