**UCS503- Software Engineering Lab**

**AUTOMATIC BRAIN TUMOR DETECTION AND CLASSIFICATION ON MRI IMAGES USING MACHINE LEARNING TECHNIQUES**

**UCS503 Software Engineering Project Report**

**End-Semester Evaluation**

**Submitted by:**

**102003759 SUSHANT VIJ**

**102003766 SAHIL CHHABRA**

**102003767 SAKET KANDHARI**

**102183051 SARTHAK TIWARI**

**Group No:**

**3COE30**

**Submitted to:**

**Dr. Manish Kumar**



**Computer Science and Engineering Department**

**TIET, Patiala**

**December 2022**

# ACKNOWLEDGEMENT

It is our great fortune that we have got the opportunity to carry out this project work under the supervision of **Prof. (Dr.) Manish Kumar** in the Department of Computer Science and Engineering, Thapar Institute of Engineering and Technology (TIET), Punjab, India. We express our sincere thanks and deepest sense of gratitude to our guide for his constant support, unparalleled guidance and limitless encouragement.

Sahil Chhabra 102003766

Sarthak Tiwari 102183051

Saket Kandhari 102003767

Sushant Vij 102003759

# CONTENTS

|  |  |  |
| --- | --- | --- |
| S No. | Assignment | PAGE No. |
|  | **PROJECT SELECTION PHASE** | **5** |
| 1.1 | 1. **SOFTWARE BID** 2. **INTRODUCTION** | **6-7** |
| 2. | **ANALYSIS PHASE** | **8-9** |
| 2.1 | 1. **USE CASE DIAGRAMS** 2. **SWIMLANE DIAGRAMS** 3. **DATA FLOW DIAGRAMS** 4. **SOFTWARE SPECIFICATION REQUIREMENTS** | **10-18** |
| 3. | **DESIGN PHASE** | **19-24** |
| 3.1 | 1. **CLASS DIAGRAM** 2. **SEQUENCE DIAGRAM** | **25** |
| 4. | **IMPLEMENTATION** | **26-36** |
| 5. | **TESTING** | **37-43** |
| 6. | **CONCLUSION** | **44** |
| 7. | **FUTURE SCOPE** | **45** |
| 8. | **REFERENCES** | **48** |

# INTRODUCTION

Brain tumor is one of the most rigorous diseases in the medical science. An effective and efficient analysis is always a key concern for the radiologist in the premature phase of tumor growth. Histological grading, based on a stereotactic biopsy test, is the gold standard and the convention for detecting the grade of a brain tumor. The biopsy procedure requires the neurosurgeon to drill a small hole into the skull from which the tissue is collected. There are many risk factors involving the biopsy test, including bleeding from the tumor and brain causing infection, seizures, severe migraine, stroke, coma and even death. But the main concern with the stereotactic biopsy is that it is not 100% accurate which may result in a serious diagnostic error followed by a wrong clinical management of the disease.

Tumor biopsy being challenging for brain tumor patients, non-invasive imaging techniques like Magnetic Resonance Imaging (MRI) have been extensively employed in diagnosing brain tumors. Therefore, development of systems for the detection and prediction of the grade of tumors based on MRI data has become necessary. But at first sight of the imaging modality like in Magnetic Resonance Imaging (MRI), the proper visualisation of the tumor cells and its differentiation with its nearby soft tissues is somewhat difficult task which may be due to the presence of low illumination in imaging modalities or its large presence of data or several complexity and variance of tumors-like unstructured shape, viable size and unpredictable locations of the tumor.

Automated defect detection in medical imaging using machine learning has become the emergent field in several medical diagnostic applications. Its application in the detection of brain tumor in MRI is very crucial as it provides information about abnormal tissues which is necessary for planning treatment.Studies in the recent literature have also reported that automatic computerized detection and diagnosis of the disease, based on medical image analysis, could be a good alternative as it would save radiologist time and also obtain a tested accuracy. Furthermore, if computer algorithms can provide robust and quantitative measurements of tumor depiction, these automated measurements will greatly aid in the clinical management of brain tumors by freeing physicians from the burden of the manual depiction of tum

# Planning Phase

Project Overview:

Medical images are one of the most important resources used by doctors to diagnose brain tumors. A tool with high accuracy to automate this process can be extremely valuable. However, because of issues related to legal liabilities, such a tool cannot replace the expert opinions of trained physicians. In this paper, we design a system to correctly classify new brain MRI images into images with tumor and images with-out tumor.

Objectives:

The conventional method for defect detection in magnetic resonance brain images is human inspection. This method is impractical due to large amount of data. Hence, trusted and automatic classification schemes are essential to prevent the death rate of human.

Scope:

The MRI brain tumor detection is complicated task due to complexity and variance of tumors. In this project, we propose the machine learning algorithms to overcome the drawbacks of traditional classifiers where tumor is detected in brain MRI using machine learning algorithms. Machine learning and image classifier can be used to efficiently detect cancer cells in brain through MRI.

Functional Requirements:

* Requires an appropriate dataset with clean MRI images
* Requires previously trained models through transfer learning for efficient feature detection

Non-Functional Requirements:

* Aiming for highest accuracy possible
* Can be available for actual use if high medical use accuracy can be achieved
* Challenges:
* Good quality dataset
* Appropriate knowledge about deep learning

# ANALYSIS PHASE

# Use case Diagram:

# 

# Swimlane Diagrams:

# 

# Data flow diagrams:

# *Level 0:*

# 

# *Level1:*

# 

# *Level 2:*

# 

# Software Requirements Specification:

## **Purpose of this Document:**

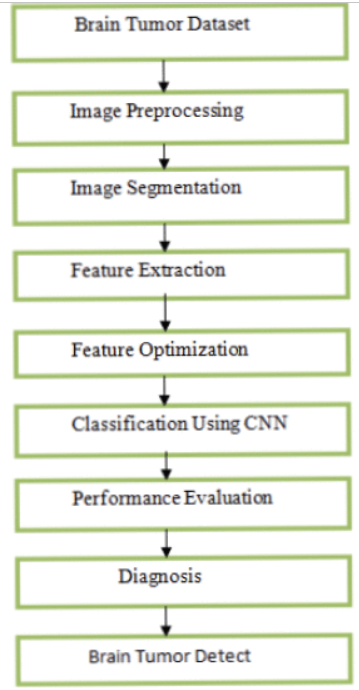
The purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. **It defines how our client, team and audience see the product and its functionality.**

**Scope of the Development Project:**

The goal is to design software early detection and prevention of brain tumour

using Transfer Learning and Convolutional Neural Network. In this system, a user will provide his/her MRI images to a hospital management having efficient devices for detection of Brain Tumor. All this data that’s being accessed by Hospital Management will be stored in a central repository (database server) and will also be replicated onto a backup database server on a daily basis so that cases of data loss are minimized in case of events such as power failure or link failure.

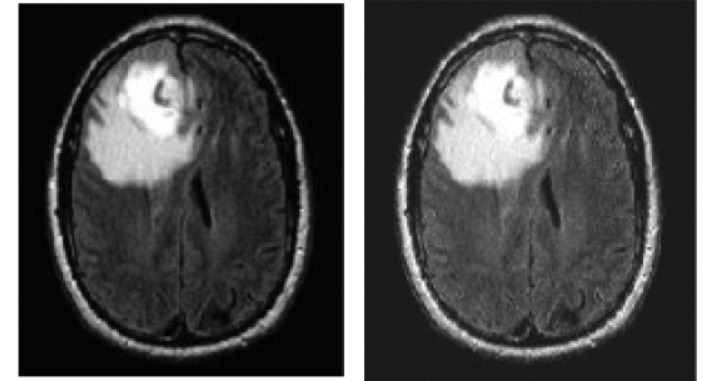
The human brain is modeled through the design and application of a neural network. These Papers confer the Brain Tumor Detection based on brain region photos on MRI Images using CNN Model. Brain regions are in the first level, extracted from the MRI image and any slice in that area is segmented to get tumors. The tumor regions that are segmented are used of CNN Architecture. CNN is used to assess the Patient Images. The primary aim of this research is to detect Brain Tumor. If the tumor present in the brain part of a patient is malignant or benign. Figure 3 shows a block diagram of a convolutional neural network-based brain tumor classification system. The preparation and assessment stages of the CNN-based brain tumor classification are separated. The number of photographs is separated into various groups by labeling them with terms like a tumor and non-tumor brain image, and so on. Preprocessing, feature extraction, and classification are all used in the training phase



The software must be able to perform the following operations:

### A. Image Preprocessing

Fig. 4 illustrates the preprocessing of the image. The following preprocessing steps were applied to every image: 1. Crop brain containing a section of the images. 2. Transform the images to the shape of (240, 240,3) as images have come from different sources. so they may have different images since images in the dataset come in different sizes. So all images need to be in the same form for feeding input to the neural network. 3. Normalization applied: to scale the pixel values to the 0-1 range



### B. Image Segmentation

It is the mechanism by which an image is split into regions with different shades, textures, brightness, contrast, and gray characteristics level. A digital grayscale picture is an input to the operation. Anomalies are the performance of the operation. The use of segmentation is to have greater data than existing in medical photos. Different processes, such as Neural Networks, the decision tree, and the algorithm based on rules and Bayesian Networks in a segment are used to obtain desired performance data [17] [25] [26]. There are many other segmentation methods.

1. *Thresholding Method:* As the name suggests, voxels above the threshold are known as belonging to the tumor.
2. *Region growing method:* Seed voxel is entered into the segmentation; from this seed, voxels that are identical are identified as belonging to the tumour.
3. *Region growing method:* Changes in the density between the edges of the voxels are seen as the limits of the tumors.

### C. Feature Selection

Feature Extraction is a method of collecting an image’s visual content. The extraction of features is the method of portraying the raw image in its reduced form to promote decision-making, such as the classification of patterns. After Segmentation of the brain, DWT is used for segmentation of MRI Images . The most important characteristics of the chained filter of low pass and high pass to derive features.

### D. Image Classification

Classification means marking the pictures according to their characteristics. The best function is defined through the application of GA, among other items. The other approach is advanced by involving the GA in three performance comparative classifiers, such as CNN and ML .

### E. Feature Optimization

Feature optimization is the combined process of feature selection and extraction which play a crucial role in brain image processing. The starting feature selection process decreases the dimensionality of the feature sets and take a minimum time for detecting. Then extract the best sets of feature raw dataset Genetic Algorithm(GA) is used

### G. Classification

Classification means labeling the image as per requirement in features. Apply on best feature is recognized by the Genetic Algorithm and such classifier like CNN for result

### H. Convolution Neural Network Architecture

In the area of medical image processing, the Neural Network is commonly used. Many researchers have tried to develop a model over the years that can more accurately recognize the tumor. To determine the utility of the proposed brain tumor classification system, training accuracy, validation accuracy, and validation loss are measured. We attempted to come up with an example that could correctly identify the tumor from 3D images of the brain MRI. This tumor can be identified by a fully-connected neural network, but we adopted CNN for our model because of parameter sharing and connection sparsity.

**Feasibility Report:**

Technical Feasibility:

In Technical Feasibility current resources both hardware software along with required technology are analyzed/assessed to develop project.

Our project is a feature detection algorithm that requires :

* Input data of specific format
* Availability of MRI image dataset
* Good understanding of python
* Appropriate knowledge of CNN
* Pre-trained transfer learning models

Economic Feasibility:

In Economic Feasibility study cost and benefit of the project are analyzed. The software can prove to be very cost effective as there is no human involvement during analysis of MRI images. Also, the cost of the software can be common worldwide whereas in-person checkups prices can vary drastically.

Legal Feasibility:

 Legal Feasibility Study is the study to know if proposed project conform legal and ethical requirements. For our project to confirm legal and ethical requirements it is important to remove the cases falling under false negatives during the evaluation of confusion matrix. Also, data security is another challenge as any user’s data in the wrong hands can prove to be a vulnerability.

Operational Feasibility:

In a traditional way of identification of MRI images it took a long time from the generation of MRI images to the detection of tumour. Also the MRI brain tumor detection is complicated task due to complexity and variance of tumors. In this project, we propose the machine learning algorithms to overcome the drawbacks of traditional classifiers where tumor is detected in brain MRI using machine learning algorithms. Machine learning and image classifier can be used to efficiently detect cancer cells in brain through MRI.

Scheduling Feasibility:

The time required for the analysis of MRI images is reduced as there is no involvement of checkup appointments, there is no human involvement which reduces errors, increases accuracy and moreover less time is required. The software can run 24/7 whereas there is a limitation to human efficiency. If trained appropriately the time required by the model to predict can be reduced up to nanoseconds.

**Specific Requirements :**

Project Overview:

Medical images are one of the most important resources used by doctors to diagnose brain tumors. A tool with high accuracy to automate this process can be extremely valuable. However, because of issues related to legal liabilities, such a tool cannot replace the expert opinions of trained physicians. In this paper, we design a system to correctly classify new brain MRI images into images with tumor and images with-out tumor.

Objectives:

The conventional method for defect detection in magnetic resonance brain images is human inspection. This method is impractical due to large amount of data. Hence, trusted and automatic classification schemes are essential to prevent the death rate of human.

Scope:

The MRI brain tumor detection is complicated task due to complexity and variance of tumors. In this project, we propose the machine learning algorithms to overcome the drawbacks of traditional classifiers where tumor is detected in brain MRI using machine learning algorithms. Machine learning and image classifier can be used to efficiently detect cancer cells in brain through MRI.

Functional Requirements:

* Requires an appropriate dataset with clean MRI images
* Requires previously trained models through transfer learning for efficient feature detection

Non-Functional Requirements:

* Aiming for highest accuracy possible
* Can be available for actual use if high medical use accuracy can be achieved
* Challenges:
* Good quality dataset
* Appropriate knowledge about deep learning
* Improving Accuracy

# DESIGN PHASE

# Class Diagram:

# 

# Sequence Diagram

# 

# ENTITY-RELATIONSHIP DIAGRAM:

# 

# IMPLEMENTATION:

### Working Theory of our Project:

**Artificial intelligence (AI):**

It is the simulation of human intelligence processes by machines, especially computer systems enabling it to even mimic human behaviour. Its applications lie in fields of Computer Vision, Natural Language Processing, Robotics, Speech Recognition, etc. Advantages of using AI are improved customer experience, accelerate speed to market, develop sophisticated products, enable cost optimisation, enhance employee productivity and improve operational efficiency. Machine Learning (ML) is a subset of AI which is programmed to think on its own, perform social interaction, learn new information from the provided data and adapt as well as improve with experience. Although training time via Deep Learning (DL) methods is more than Machine Learning methods, it is compensated by higher accuracy in the former case. Also, DL being automatic, large domain knowledge is not required for obtaining desired results unlike in ML.

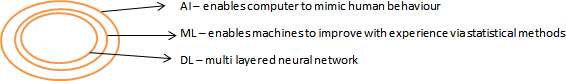


Fig: A diagram showing the sub-classes of Artificial Intelligence

### Brain tumor:

In medical science, an anomalous and uncontrollable cell growth inside the brain is recognised as tumor. Human brain is the most receptive part of the body. It controls muscle movements and interpretation of sensory information like sight, sound, touch, taste, pain, etc.

The human brain consists of Grey Matter (GM), White Matter (WM) and Cerebrospinal Fluid (CSF) and on the basis of factors like quantification of tissues, location of abnormalities, malfunctions & pathologies and diagnostic radiology, a presence of tumor is identified. A tumor in the brain can affect such sensory information and muscle movements or even results in more dangerous situation which includes death. Depending upon the place of commencing, tumor can be categorised into primary tumors and secondary tumors. If the tumor is originated inside the skull, then the tumor is known as primary brain tumor otherwise if the tumor‘s initiation place is

somewhere else in the body and moved towards the brain, then such tumors are called secondary tumors.

Brain tumor can be of the following types-glioblastoma, sarcoma, metastatic bronchogenic carcinoma on the basis of axial plane. While some tumours such as meningioma can be easily segmented, others like gliomas and glioblastomas are much more difficult to localise. World Health Organisation (WHO) categorised gliomas into - HGG/high grade glioma/glioblastoma/IV stage /malignant & LGG/low grade glioma/II and III stage /benign. Although most of the LGG tumors have slower growth rate compared to HGG and are responsive to treatment, there is a subgroup of LGG tumors which if not diagnosed earlier and left untreated could lead to GBM. In both cases a correct treatment planning (including surgery, radiotherapy, and chemotherapy separately or in combination) becomes necessary, considering that an early and proper detection of the tumor grade can lead to a good prognosis. Survival time for a GBM (Glioblastoma Multiform) or HGG patient is very low i.e. in the range of 12 to 15 months.

Magnetic Resonance Imaging (MRI) has become the standard non-invasive technique for brain tumor diagnosis over the last few decades, due to its improved soft tissue contrast that does not use harmful radiations unlike other methods like CT(Computed Tomography), X-ray, PET (Position Emission Tomography) scans etc. The MRI image is basically a matrix of pixels having characteristic features.

Since glioblastomas are inﬁltrative tumours, their borders are often fuzzy and hard to distinguish from healthy tissues. As a solution, more than one MRI modality is often employed

e.g. T1 (spin-lattice relaxation), T1-contrasted (T1C), T2 (spin-spin relaxation), proton density (PD) contrast imaging, diffusion MRI (dMRI), and ﬂuid attenuation inversion recovery (FLAIR) pulse sequences. T1-weighted images with intravenous contrast highlight the most vascular regions of the tumor (T 1C gives much more accuracy than T1.), called ‗Enhancing tumor‘ (ET), along with the ‗tumor core' (TC) that does not involve peritumoral edema. T2-weighted (T2W) and T2W-Fluid Attenuation Inversion Recovery (FLAIR) images are used to evaluate the tumor and peritumoral edema together defined as the ‗whole tumor‘ (WT). Gliomas and glioblastomas are difficult to distinguish in T1, T1c, T2 and PD. They are better identified in FLAIR modalities.

We have attempted to separate the brain tumor into following types-necrosis (1), edema (2), non- enhancing (malignant) (3) and enhancing (benign) (4) tumor. MRI images can be of three types on the basis of position from which they are taken which are Sagittal (side), Coronal (back) and Axial (top). We have used sagittal images in our project.

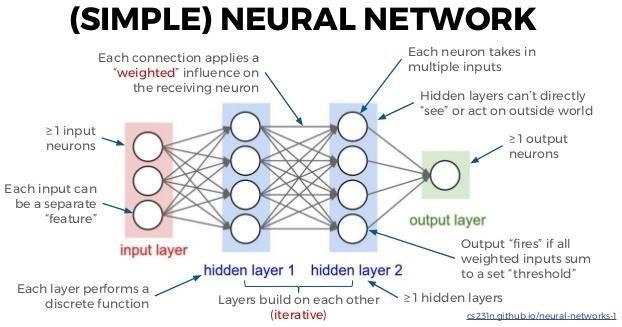
Process of brain tumor segmentation can be manual selection of ROI, Semi-automatic and fully-automatic. Popular machine learning algorithms for classification of brain tumor are Artificial Neural Network, Convolutional Neural Network, k-Nearest Neighbour (kNN), Decision Tree, Support Vector Machine (SVM), Naïve Bayes and Random Field (RF). Here, we are using Convolutional Neural Network (CNN) for the detection and classification of the brain tumor.

### Basic Operation of Neural Networks:

Neural Networks (NN) form the base of deep learning, a subfield of machine learning where the algorithms are inspired by the structure of the human brain. NN take in data, train themselves to recognize the patterns in this data and then predict the outputs for a new set of similar data. NN are made up of layers of neurons. These neurons are the core processing units of the network. First we have the input layer which receives the input; the output layer predicts our final output. In between, exist the hidden layers which perform most of the computations required by our network.

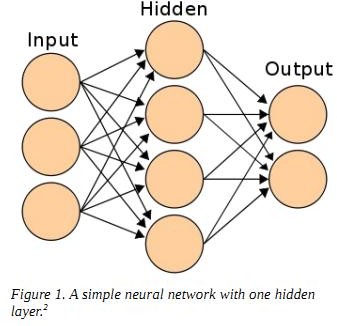
Our brain tumor images are composed of 128 by 128 pixels which make up for 16,384 pixels. Each pixel is fed as input to each neuron of the first layer. Neurons of one layer are connected to neurons of the next layer through channels .Each of these channels is assigned a numerical value known as ‗weight‘. The inputs are multiplied to the corresponding weight and their sum is sent as input to the neurons in the hidden layer. Each of these neurons is associated with a numerical value called the ‗bias‘ which is then added to the input sum. This value is then passed through a threshold function called the ‗activation function‘. The result of the activation function determines if the particular neuron will get activated or not. An activated neuron transmits data to the neurons of the next layer over the channels. In this manner the data is propagated through the network this is called ‗forward propagation‘. In the output layer the neuron with the highest value fires and determines the output. The values are basically a probable. The predicted output is compared against the actual output to realize the ‗error‘ in prediction. The magnitude of the

error gives an indication of the direction and magnitude of change to reduce the error. This information is then transferred backward through our network. This is known as ‗back propagation‘. Now based on this information the weights are adjusted. This cycle of forward propagation and back propagation is iteratively performed with multiple inputs. This process continues until our weights are assigned such that the network can predict the type of tumor correctly in most of the cases. This brings our training process to an end. NN may take hours or even months to train but time is a reasonable trade-off when compared to its scope Several experiments show that after pre-processing MRI images, neural network classification algorithm was the best more specifically CNN(Convolutional Neural Network) as compared to Support Vector Machine(SVM),Random Forest Field.



Input from medical professionals or users

Fig: A multi-layer perceptron model of neural network



Weights(w)

Nodes

Results are shown on IoT based devices or Web-based applications

### Transfer Learning:

A major assumption in many machine learning and data mining algorithms is that the training and future data must be in the same feature space and have the same distribution. However, in many real-world applications, this assumption may not hold. For example, we sometimes have a classification task in one domain of interest, but we only have sufficient training data in another domain of interest, where the latter data may be in a different feature space or follow a different data distribution. In such cases, knowledge transfer, if done successfully, would greatly improve the performance of learning by avoiding much expensive data labelling efforts. In recent years, transfer learning has emerged as a new learning framework to address this problem.

Transfer learning allows neural networks using significantly less data .With transfer learning, we are in effect transferring the ‗knowledge‘ that a model has learned from a previous task, to our current one. The idea is that the two tasks are not totally disjoint, as such we can leverage whatever network parameters that model has learned through its extensive training, without having to do that training ourselves. Transfer learning has been consistently proven to boost model accuracy and reduce required training time, less data, less time, more accuracy.

Transfer learning is classified to three different settings: inductive transfer learning, transductive transfer learning and unsupervised transfer learning. Most previous works focused on the settings. Furthermore, each of the approaches to transfer learning can be classified into four contexts based on ―what to transfer‖ in learning. They include the instance-transfer approach, the feature-representation-transfer approach, the parameter transfer approach and the relational- knowledge-transfer approach, respectively.

The smaller networks converged & were then used as initializations for the larger, deeper networks- This process is called pre-training. While making logical sense, pre-training is a very time consuming, tedious task, requiring an entire network to be trained before it can serve as an initialization for a deeper network.

### Activation Function:

Sigmoid function ranges from 0 to 1 and is used to predict probability as an output in case of binary classification while Softmax function is used for multi-class classification. tanh function ranges from -1 to 1 and is considered better than sigmoid in binary classification using feed forward algorithm. ReLU (Rectified Linear Unit) ranges from 0 to infinity and Leaky ReLU

(better version of ReLU) ranges- from -infinity to +infinity. ReLU stands for Rectified Linear Unit for a non-linear operation. The output is ***ƒ(x) = max(0,x).***ReLU‘s purpose is to introduce non-linearity in our ConvNet. Since, the real world data would want our ConvNet to learn would be non-negative linear values. There are other nonlinear functions such as tanh or sigmoid that can also be used instead of ReLU. Most of the data scientists use ReLU since performance wise ReLU is better than the other two.

Stride is the number of pixels that would move over the input matrix one at a time.

Sometimes filter does not fit perfectly fit the input image. We have two options: either pad the picture with zeros (zero-padding) so that it fits or drop the part of the image where the filter did not fit. This is called valid padding which keeps only valid part of the image.

### Convolutional Neural Network:

Classifier models can be basically divided into two categories respectively which are generative models based on hand- crafted features and discriminative models based on traditional learning such as support vector machine (SVM), Random Forest (RF) and Convolutional Neural Network (CNN). One difficulty with methods based on hand-crafted features is that they often require the computation of a large number of features in order to be accurate when used with many traditional machine learning techniques. This can make them slow to compute and expensive memory-wise. More efficient techniques employ lower numbers of features, using dimensionality reduction like PCA (Principle Component Analysis) or feature selection methods, but the reduction in the number of features is often at the cost of reduced accuracy. Brain tumour segmentation employ discriminative models because unlike generative modelling approaches, these approaches exploit little prior knowledge on the brain‘s anatomy and instead rely mostly on the extraction of [a large number of] low level image features, directly modelling the relationship between these features and the label of a given voxel.

In our project, we have used the Convolutional Neural Network architecture for Brain tumor Detection and Classification.

Convolutional neural network processes closely knitted data used for image classification, image processing, face detection etc. It is a specialised 3D structure with specialised NN analysing RGB layers of an image .Unlike others, it analyses one image at a time

,identifies and extracts important features and uses them to classify the image .Convolutional

Neural Networks (ConvNets) automatically learns mid-level and high-level representations or abstractions from the input training data. The main building block used to construct a CNN architecture is the convolutional layer. It also consists of several other layers, some of which are described as bellow:

* Input Layer-It takes in the raw pixel value of input image
* Convolutional Layer- It is the first layer to extract features from an input image. Convolution preserves the relationship between pixels by learning image features using small squares of input data. It is a mathematical operation that takes two inputs such as image matrix and a filter or kernel to generate a feature map Convolution of an image with different filters can perform operations such as edge detection, blur and sharpen by applying filters.
* Activation Layer-It produces a single output based on the weighted sum of inputs
* Pooling Layer-Pooling layers section would reduce the number of parameters when the images are too large. Spatial pooling (also called subsampling or down sampling) reduces the dimensionality of each map but retains important information. Spatial pooling can be of different types:
  + Max Pooling – taking the largest element in the feature map
  + Average Pooling - taking the average of elements in the feature map
  + Sum Pooling – taking the sum of all elements in the feature map
* Fully Connected Layer-The layer we call as FC layer, we flattened our matrix into vector and feed it into a fully connected layer like a neural network. the feature map matrix will be converted as column vector (x1, x2, x3, …). With the fully connected layers, we combined these features together to create a model. Forclassifying input image into various classes based on training set.
* Dropout Layer-It prevents nodes in a network from co-adapting to each other.

#### Advantages-

* 1. It is considered as the best ml technique for image classification due to high accuracy.
  2. Image pre-processing required is much less compared to other algorithms.
  3. It is used over feed forward neural networks as it can be trained better in case of complex images to have higher accuracies.
  4. It reduces images to a form which is easier to process without losing features which are critical for a good prediction by applying relevant filters and reusability of weights
  5. It can automatically learn to perform any task just by going through the training data i.e. there no need for prior knowledge
  6. There is no need for specialised hand-crafted image features like that in case of SVM, Random Forest etc.

#### Disadvantages-

1. It requires a large training data.
2. It requires appropriate model.
3. It is time consuming.
4. It is a tedious and exhaustive procedure.
5. While convolutional networks have already existed for a long time, their success was limited due to the size of the considered network.

**Solution**-Transfer Learning for inadequate data which will replace the last fully connected layer with pre-trained ConvNet with new fully connected layer.



Input Layer

Activation Layer

Convolutional Layer

Batch Normalisation Layer

Max Pooling Layer





Output Layer

Dense Layer

Flatten Layer

Dropout Layer

Fully Connected Layer

Fig: A diagram of a model trained from scratch using CNN architecture.

### Evaluation Metrics:

* True Positive (TP) is the HGG class predicted in the presence of the LGG class of the glioma. True Negative (TN) is the LGG class predicted in the absence of the HGG class of glioma. False Positive (FP) is prediction of HGG class in the absence of LGG class. False Negative (FN) is prediction of LGG class in the absence of HGG class.
* Accuracy is the most intuitive performance measure. Accuracy is the amount of correctly

prediction made by the total number of predictions made. Accuracy = 𝑇𝑃+𝑇𝑁

𝑇𝑃+𝐹𝑃+𝑇𝑁+𝐹𝑁

* Precision is defined as the number of true positives divided by the number of true

positives plus the number of false positives.Precision = 𝑇𝑃

𝑇𝑃+𝐹𝑃

* Recall is also known as sensitivity. It is the fraction of the total amount of relative

relevant instances that were actually retrieved.Recall = 𝑇𝑃

𝑇𝑃+𝐹𝑁

* F 1 Score is the weighted average or the harmonic mean of Precision and Recall taking

both metrics into account in the following equation: F1 Score = 2 x 𝑝𝑟𝑒𝑐𝑖𝑠𝑖𝑜𝑛 ∗ 𝑟𝑒𝑐𝑎𝑙𝑙

𝑝𝑟𝑒𝑐𝑖𝑠𝑖𝑜𝑛 +𝑟𝑒𝑐𝑎𝑙𝑙

.When we have an unbalanced dataset F 1 Score favoured over accuracy because it takes both false positives and false negatives into account. F-measures are used to balance the

ratio of false negatives using a weighting parameter (beta) it is given as F = 𝑃 ∗ 𝑅 (1+𝛽)2

(𝑃+𝑅)𝛽 2

* Other performance metrics used are: sensitivity, specificity and error rate. Sensitivity represents the probability of predicting actual HGG class. Specificity value defines prediction of LGG class. They allow us to determine potential of over- or under- segmentations of the tumor sub-regions. The error rate (ERR) is the amount of predicted class that have been incorrectly classified by a decision model. The overall classification is also provided by the Area under the Curve (AUC) that represents better classification if the area under the curve is more. All of these performances metric is evaluated for FLAIR sequences.
* The DSC(dice similarity co-efficient) measures the overlap between the manual delineated brain tumour regions and the segmentation results of our fully automatic method that is. Mathematically, dice score/DSC is the number of false positives divided

by the number of positives added with the number of false positives. DSC = 2𝑇𝑃

𝐹𝑃+𝑇𝑃+𝐹𝑁

and Dice loss = 2|𝑋1⊓𝑌1|

|𝑋1|+|

# METHODOLOGY:

### Software Requirements:

Python 3 - We have used Python which is a statistical mathematical programming language like R instead of MATLAB due to the following reasons:

1. Python code is more compact and readable than MATLAB
2. The python data structure is superior to MATLAB
3. It is an open source and also provides more graphic packages and data sets

Keras (with TensorFlow backend 2.3.0 version) - Keras is a neural network API consisting of TensorFlow, CNTk, Theano etc.

Python packages like Numpy, Matplotlib, Pandas for mathematical computation and plotting graphs, SimpleITK for reading the images which were in .mha format and Mahotas for feature extraction of GLCM

Kaggle was used to obtain the online dataset.

GitHub and Stackoverflow was used for reference in case of programming syntax errors.

OpenCV (Open Source Computer Vision) is a library of programming functions aimed at real time computer vision i.e. used for image processing and any operations relating to image like reading and writing images, modifying image quality, removing noise by using Gaussian Blur, performing binary thresholding on images, converting the original image consisting of pixel values into an array, changing the image from RGB to grayscale etc. It is free to use, simple to learn and supports C++, Java, C, Python. Its popular application lies in CamScanner or Instagram, GitHub or a web-based control repository.

Google Colaboratory (open-source Jupyter Notebook interface with high GPU facility) - Google Colab /Colaboratory is a free Jupyter notebook environment that requires no setup and runs entirely on cloud. With Colab, one can write and execute code, save and share analyses, access powerful computing resources, all for free from browser.[Jupyter Notebook is a powerful way to iterate and write on your Python code for data analysis. Rather than writing and rewriting an entire code, one can write lines of code and run them at a time. It is built off of iPython which

is an interactive way of running Python code. It allows Jupyter notebook to support multiple languages as well as storing the code and writing own markdown.]

### Hardware Requirements:

Processor: Intel® Core™ i3-2350M CPU @ 2.30GHz Installed memory (RAM):4.00GB

System Type: 64-bit Operating System

### Image Acquisition:

**Kaggle dataset:**

Images can be in the form of .csv (comma separated values), .dat (data) files in grayscale, RGB, or HSV or simply in .zip file as was in the case of our online Kaggle dataset. It contained 98 healthy MRI images and 155 tumor infected MRI images.

### Br35H dataset:

The dataset contains 3 folders: yes, no and pred which contains 3060 Brain MRI Images.

| Folder | Description |
| --- | --- |
| Yes | The folder yes contains 1500 Brain MRI Images that are tumorous |
| No | The folder no contains 1500 Brain MRI Images that are non-tumorous |

### Data Augmentation:

Data augmentation consists of Grey Scaling(RGB/BW to ranges of grey),Reflection(vertical/horizontal flip),Gaussian Blur(reduces image noise),Histogram equalisation(increases global contrast),Rotation(may not preserve image size),Translation(moving the image along x or y axis), linear transformation such as random rotation (0-10 degrees), horizontal and vertical shifts, and horizontal and vertical ﬂips. Data Augmentation is done to

teach the network desired invariance and robustness properties, when only few training samples are available.

### Image Pre-Processing:

Our pre-processing includes rescaling, noise removal to enhance the image, applying Binary Thresholding and morphological operations like erosion and dilation, contour forming (edge based methodology). In the first step of pre-processing, the memory space of the image is reduced by scaling the gray-level of the pixels in the range 0-255. We used Gaussian blur filter for noise removal as it is known to give better results than Median filter since the outline of brain is not segmented as tumor here.

### Segmentation:

Brain tumor segmentation involves the process of separating the tumor tissues (Region of Interest – ROI) from normal brain tissues and solid brain tumor with the help of MRI images or other imaging modalities. Its mechanism is based on identifying similar type of subjects inside an image and forms a group of such by either finding the similarity measure between the objects and group the objects having most similarity or finding the dissimilarity measure among the objects and separate the most dissimilar objects in the space. Segmentation algorithms can be of two type which are bi-clusters (2 sub-parts) or multi-clustered (more than 2 sub-parts) algorithms. Segmentation can be done by using-Edge Detection, Region Growing, Watershed, Clustering via FCM, Spatial Clustering, Split and Merge Segmentation and Neural Network via MLP(ANN+DWT).

In order to identify the tumor region from the brain image, Binary Thresholding can be used (via Region Growing method), which converts a gray scale image to binary image based on the selected threshold values. The problems associated with such approach are that binary image results in loss of texture and the threshold value comes out be different for different images. Hence, we are looking for a more advanced segmentation algorithm, the watershed algorithm by using Otsu Binarisation.

### Feature Extraction:

Feature Extraction is the mathematical statistical procedure that extracts the quantitative parameter of resolution changes/abnormalities that are not visible to the naked eye. Examples of

such features are Entropy, RMS, Smoothness, Skewness, Symmetry, Kurtosis, Mean, Texture, Variance, Centroid, Central Tendency, IDM (Inverse Difference Moment

),Correlation,Energy,Homogeneity,Dissimilarity,Contrast,Shade,Prominence,Eccentricity, Perimeter, Area and many more.

Feature Extraction is identifying abnormalities. We need to extract some features from images as we need to do classification of the images using a classifier which needs these features to get trained on. We chose to extract GLCM (texture-based features). Gray Level Co-occurrence Matrix (GLCM) features are based on probability density function and frequency of occurrence of similar pixels. GLCM is a statistical method of examining texture that considers the spatial relationship of pixels.

### Machine Learning Training and Testing:

Models for image classification with weights on ImageNet are Xception,VGG16,VGG19,ResnNet,ResNet2, ResNet 50, Inception v2, Inception v3, MobileNet, MobileNet v2, ,DenseNet, AlexNet, GoogleNet, NasNet etc. For the implementation of Transfer Learning in our project, we have chosen VGG16, ResNet50 and Inception v3 as out samples.

After training the model, we need to validate and fine-tune the parameters and finally test the model on unknown samples where the data undergoes feature extraction on the basis of which the model can predict the class by matching corresponding labels. To achieve this, we can either split our dataset in the ratio of -60/20/20 or 70/20/10. We have used the former one.

For a given training dataset, back-propagation learning may proceed in one of the following two basic ways:

* Pattern/Sequential/Incremental mode where the whole sequence of forward and backward computation is performed resulting in weight adjustment for each pattern. It again starts from the first pattern till errors are minimised, within acceptable levels. It is done online, requires less local storage, faster method and is less likely to be trapped in local minima.
* Batch mode where the weight upgradation is done after all the N training sets or ‗epochs‘ are presented. After presentation of the full set, weights are upgraded and then again the whole batch/set is presented iteratively till the minimum acceptable error is arrived at by comparing the target and actual outputs. Training stops when a given number of epochs elapse or when the error reaches an acceptable level or when the error stops improving.

In supervised network, the network learns by comparing the network output with the correct answer. The network receives feedback about the errors by matching the corresponding labels and weights in different layers and adjusts its weights to minimise the error. It is also known as learning through teacher or ‗Reinforced Learning‘.

In unsupervised network, there is no teacher i.e. labels are not provided along with the data to the network. Thus, the network does not get any feedback about the errors. The network itself discovers the interesting categories or features in the input data. In many situations, the learning goal is not known in terms of correct answers. The only available information is in the correlation of input data or signals. The unsupervised networks are expected to recognise the input patterns, classify these on the basis of correlations and produce output signals corresponding to input categories. It is a type of dynamic programming that trains algorithm using a system of reward and punishment. Agent learns without human interaction and examples and only by interacting with the environment. For our purpose, we have used supervised network or Reinforced Learning for training our model.

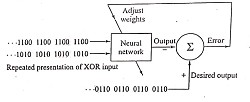
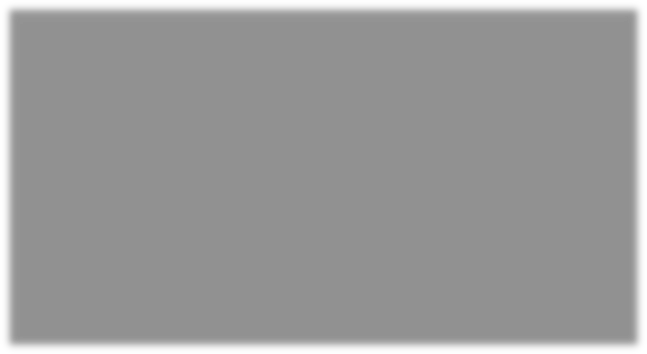
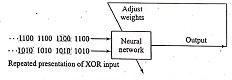
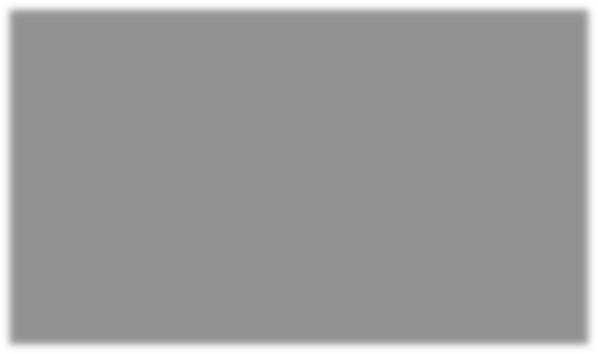


Fig: A diagram showing Unsupervised (left) and Supervised Learning Network (right)

# Flowchart for Design And Development of Project

Data Collection

Image pre-processing

Segmentation via binary thresholding

Feature extraction

Model construction

Machine Learning training

Tumor detection and classification

Validation on unknown test samples

Analysis and Conclusion

**Python Program for the Project:**

* Training the model :

import cv2

import os

from PIL import Image

import numpy as np

image\_directory='./datasets/'

no\_tumor\_images = os.listdir(image\_directory+'no/')     #lists all images in no directory

yes\_tumor\_images = os.listdir(image\_directory+'yes/')

dataset = []

label = []

# print(no\_tumor\_images)

path = 'no0.jpg'

# print(path.split('.'))      #['no0', 'jpg']

path.split('.')[1]

for i, image\_name in enumerate(no\_tumor\_images):

    if(image\_name.split('.')[1]=='jpg'):

        image = cv2.imread(image\_directory+'no/'+image\_name)

        image = Image.fromarray(image, 'RGB')

        image = image.resize((64,64))

        dataset.append(np.array(image))

        label.append(0)

for i, image\_name in enumerate(yes\_tumor\_images):

    if(image\_name.split('.')[1]=='jpg'):

        image = cv2.imread(image\_directory+'yes/'+image\_name)

        image = Image.fromarray(image, 'RGB')

        image = image.resize((64,64))

        dataset.append(np.array(image))

        label.append(1)

print(len(label))

print(len(dataset))

#converting dataset into numpy array:

dataset = np.array(dataset)

label = np.array(label)

from sklearn.model\_selection import train\_test\_split

x\_train, x\_test, y\_train, y\_test = train\_test\_split(dataset, label, test\_size = 0.2, random\_state= 0)

print(x\_train.shape)

import tensorflow as tf

from tensorflow import keras

from tensorflow.keras.utils import normalize

# x\_test = x\_test/255

x\_train = normalize(x\_train, axis = 1)

x\_test = normalize(x\_test, axis = 1)

# x\_train = x\_train/255

#building our model

from keras.models import Sequential

from keras.layers import Conv2D, MaxPooling2D, Activation

from keras.layers import Dropout, Flatten, Dense

# Model Building

model = Sequential()

model.add(Conv2D(32, (3, 3), input\_shape = (64,64,3)))

model.add(Activation('relu'))

model.add(MaxPooling2D(pool\_size = (2,2)))

model.add(Conv2D(64, (3, 3), kernel\_initializer='he\_uniform'))

model.add(Activation('relu'))

model.add(MaxPooling2D(pool\_size = (2,2)))

model.add(Conv2D(32, (3, 3), kernel\_initializer='he\_uniform'))

model.add(Activation('relu'))

model.add(MaxPooling2D(pool\_size = (2,2)))

model.add(Flatten())

model.add(Dense(64))

model.add(Activation('relu'))

model.add(Dropout(0.5))

model.add(Dense(1))

model.add(Activation('sigmoid'))    #categorical cross entropy softmax

model.compile(loss = 'binary\_crossentropy', optimizer = 'adam', metrics = ['accuracy'])

model.fit(x\_train, y\_train,

          batch\_size = 16,          #2400/16 = 150

          verbose = 1,

          epochs = 10,

          validation\_data = (x\_test, y\_test),

          shuffle = False)

model.save('BrainTumor10Epochs.h5')         #test\_accuracy => 97.17% train\_accuracy => 98.96%

# Testing :

import cv2

from keras.models import load\_model

model = load\_model('BrainTumor10Epochs.h5')

image = cv2.imread('.\pred\pred56.jpg')

import matplotlib.pyplot as plt

import matplotlib.image as mpimg

from PIL import Image

img = Image.fromarray(image, 'RGB')

img = img.resize((64,64))

imgplot = plt.imshow(image)

plt.show()

import numpy as np

img = np.array(img)

print(img)

print(img.shape)

input\_img = np.expand\_dims(img, axis=0)

print(input\_img)

print(input\_img.shape)

result=model.predict(input\_img)

print(result)

# Code Transfer Learning :

import cv2

import os

from PIL import Image

import numpy as np

image\_directory='./datasets/'

no\_tumor\_images = os.listdir(image\_directory+'no/')     #lists all images in no directory

yes\_tumor\_images = os.listdir(image\_directory+'yes/')

dataset = []

label = []

# print(no\_tumor\_images)

path = 'no0.jpg'

# print(path.split('.'))      #['no0', 'jpg']

path.split('.')[1]

for i, image\_name in enumerate(no\_tumor\_images):

    if(image\_name.split('.')[1]=='jpg'):

        image = cv2.imread(image\_directory+'no/'+image\_name)

        image = Image.fromarray(image, 'RGB')

        image = image.resize((64,64))

        dataset.append(np.array(image))

        label.append(0)

for i, image\_name in enumerate(yes\_tumor\_images):

    if(image\_name.split('.')[1]=='jpg'):

        image = cv2.imread(image\_directory+'yes/'+image\_name)

        image = Image.fromarray(image, 'RGB')

        image = image.resize((64,64))

        dataset.append(np.array(image))

        label.append(1)

print(len(label))

print(len(dataset))

#converting dataset into numpy array:

dataset = np.array(dataset)

label = np.array(label)

from sklearn.model\_selection import train\_test\_split

x\_train, x\_test, y\_train, y\_test = train\_test\_split(dataset, label, test\_size = 0.2, random\_state= 0)

import tensorflow as tf

from tensorflow import keras

from tensorflow.keras.utils import normalize

# x\_test = x\_test/255

x\_train = normalize(x\_train, axis = 1)

x\_test = normalize(x\_test, axis = 1)

# x\_train = x\_train/255

from keras.applications.vgg16 import VGG16

from keras.layers import Input, Dense, Flatten

from keras.models import Model

vgg = VGG16(input\_shape = (64,64,3),weights = 'imagenet',include\_top = False)

for layer in vgg.layers:

    layer.trainable = False

x = Flatten()(vgg.output)

prediction = Dense(1,activation = 'softmax')(x)     #one final classes

model = Model(inputs = vgg.input,outputs = prediction)

model.summary()

from tensorflow.keras.optimizers import RMSprop

model.compile(loss = 'binary\_crossentropy', optimizer = RMSprop(lr=0.0001), metrics = ['accuracy'])

model.fit(x\_train, y\_train,

          batch\_size = 16,          #2400/16 = 150

          epochs = 2,

          validation\_data = (x\_test, y\_test),

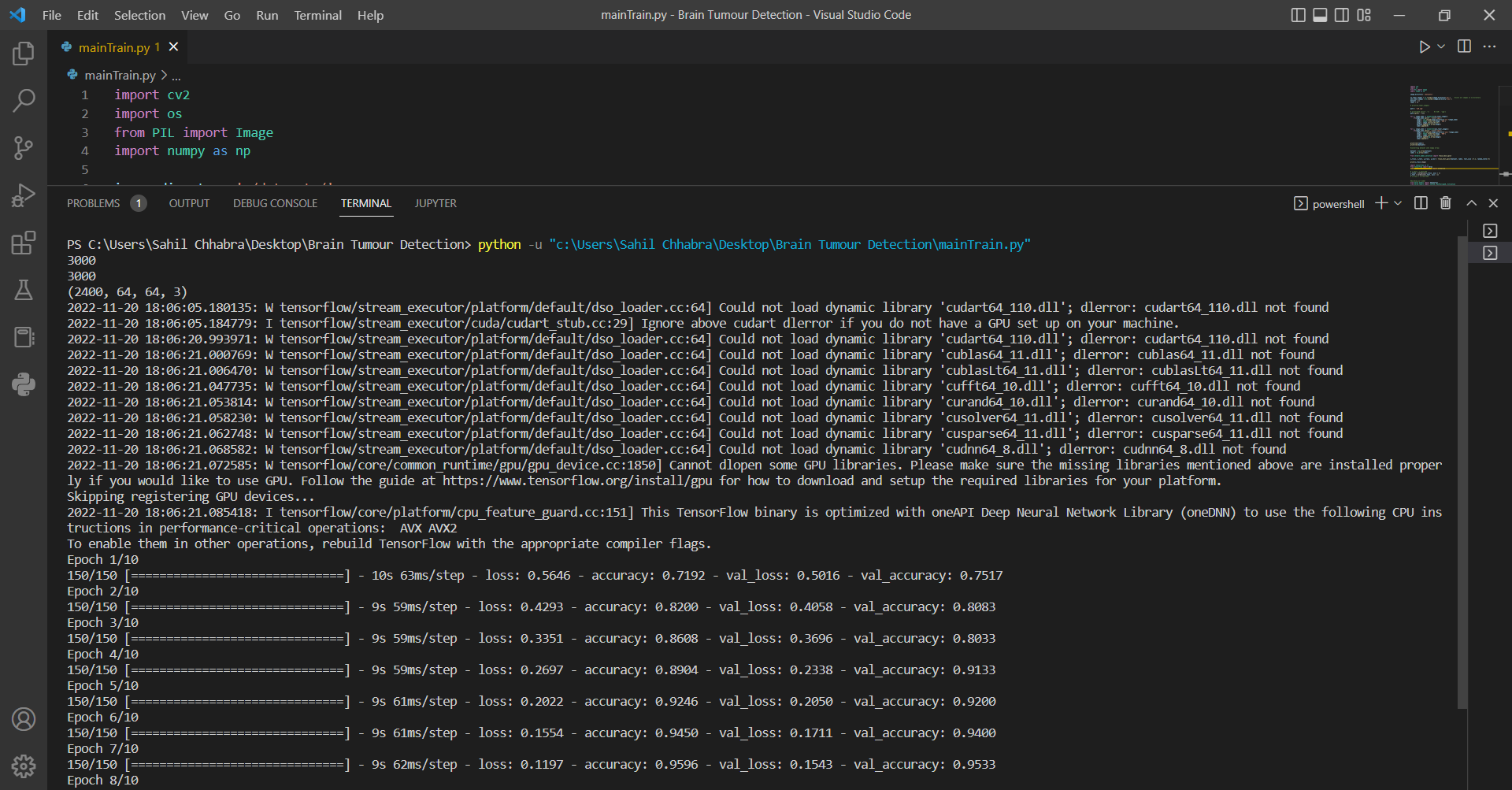
          shuffle = False)

model.save('BrainTumor10EpochsTransferL.h5')

**TESTING THE MODEL**

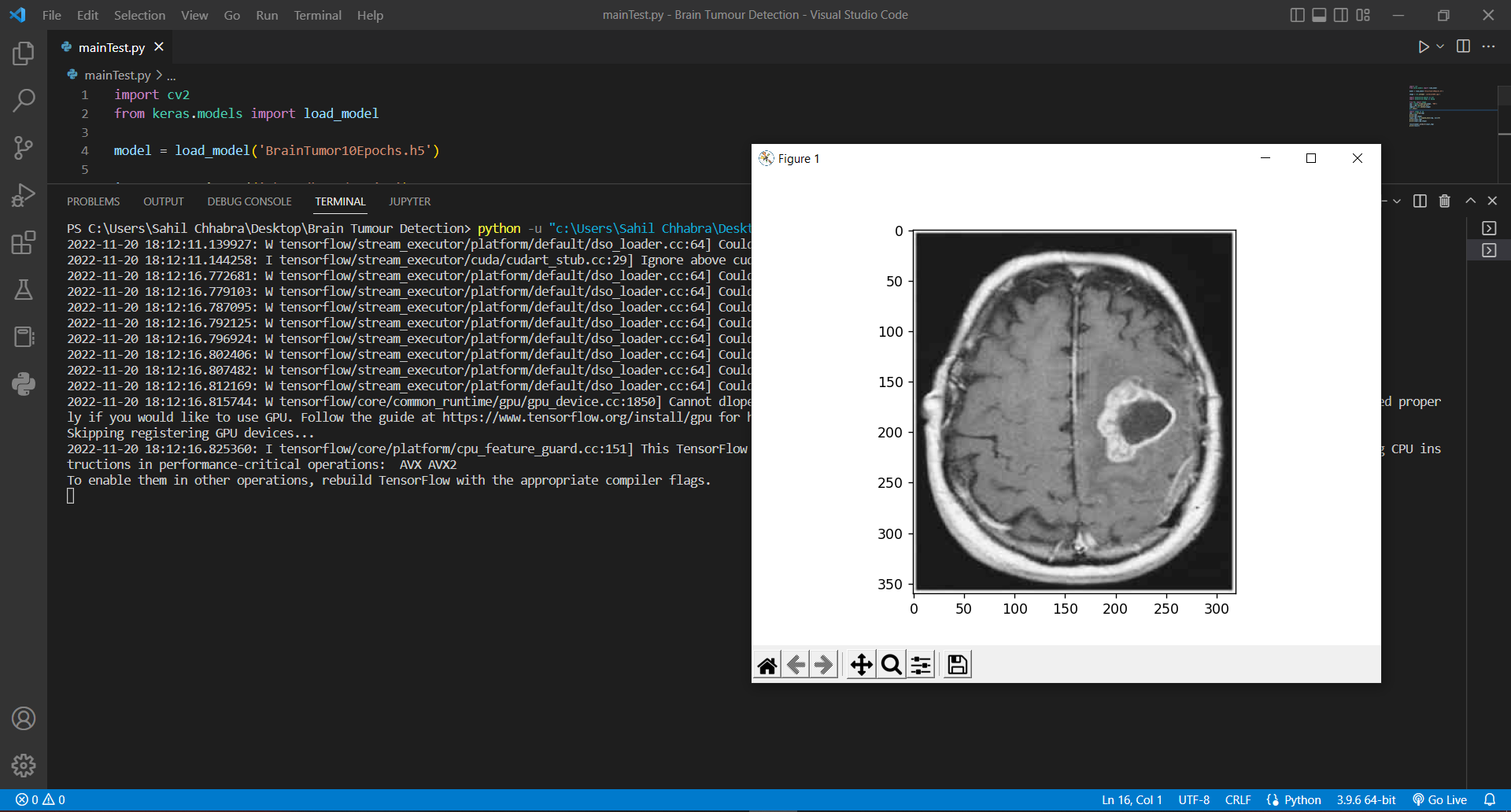
**Output Screenshots:**

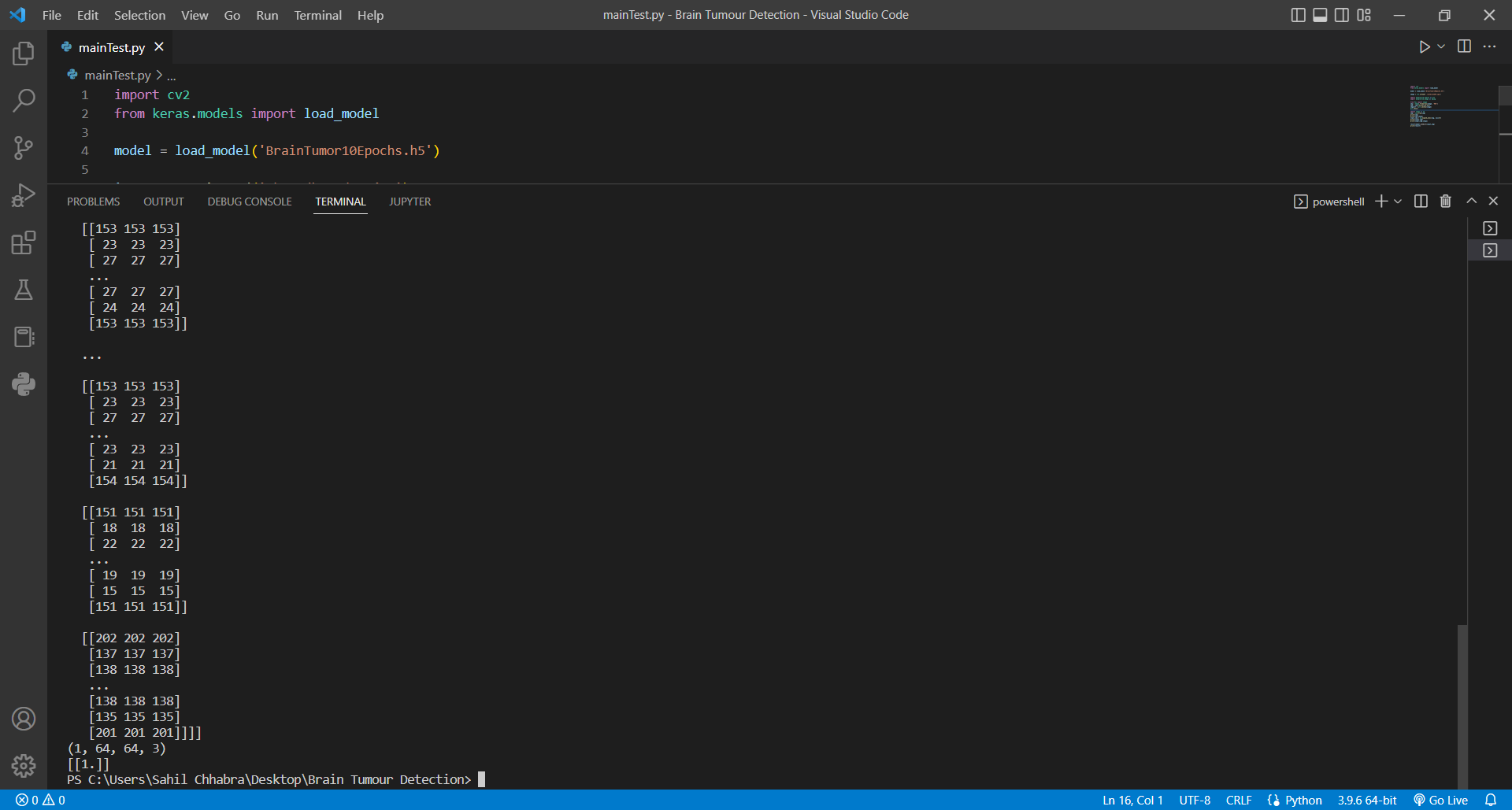
* **Training :**





* **Testing:**





# Transfer Learning :

# 

# 

# 

# 

# CONCLUSION

Without pre-trained Keras model, the train accuracy is 97.5% and validation accuracy is 90.0%.The validation result had a best figure of 91.09% as accuracy.It is observed that without using pre-trained Keras model, although the training accuracy is >90%, the overall accuracy is low unlike where pre-trained model is used.

Also, when we trained our dataset without Transfer learning, the computation time was 40 min whereas when we used Transfer Learning, the computation time was 20min. Hence, training and computation time with pre-trained Keras model was 50% lesser than without.

Chances over over-fitting the dataset is higher when training the model from scratch rather than using pre-trained Keras.Keras also provides an easy interface for data augmentation.

Amongst the Keras models, it is seen that ResNet 50 has the best overall accuracy as well as F1 score.ResNet is a powerful backbone model that is used very frequently in many computer vision tasks.

Precision and Recall both cannot be improved as one comes at the cost of the other .So, we use F1 score too.

Transfer learning can only be applied if low-level features from Task 1(image recognition) can be helpful for Task 2(radiology diagnosis).

For a large dataset, Dice loss is preferred over Accuracy.

For small size of data, we should use simple models, pool data, clean up data, limit experimentation, use regularisation/model averaging ,confidence intervals and single number evaluation metric.

To avoid overfitting, we need to ensure we have plenty of testing and validation of data

i.e. dataset is not generalised. This is solved by Data Augmentation. If the training accuracy too high, we can conclude that it the model might be over fitting the dataset. To avoid this, we can monitor testing accuracy, use outliers and noise, train longer, compare variance (=train performance-test performance).

# FUTURE SCOPE

Build an app-based user interface in hospitals which allows doctors to easily determine the impact of tumor and suggest treatment accordingly

Since performance and complexity of ConvNets depend on the input data representation we can try to predict the location as well as stage of the tumor from Volume based 3D images. By creating three dimensional (3D) anatomical models from individual patients, training, planning and computer guidance during surgery is improved.

Using VolumeNet with LOPO (Leave-One-Patient-Out) scheme has proved to give a high training as well as validation accuracy(>95%).In LOPO test scheme, in each iteration, one patient is used for testing and remaining patients are used for training the ConvNets, this iterates for each patient. Although LOPO test scheme is computationally expensive, using this we can have more training data which is required for ConvNets training. LOPO testing is robust and most applicable to our application, where we get test result for each individual patient. So, if classiﬁer misclassiﬁes a patient then we can further investigate it separately.

Improve testing accuracy and computation time by using classifier boosting techniques like using more number images with more data augmentation, fine-tuning hyper parameters, training for a longer time i.e. using more epochs, adding more appropriate layers etc.. Classifier boosting is done by building a model from the training data then creating a second model that attempts to correct the errors from the first model for faster prognosis. Such techniques can be used to raise the accuracy even higher and reach a level that will allow this tool to be a significant asset to any medical facility dealing with brain tumors.

For more complex datasets, we can use U-Net architecture rather than CNN where the max pooling layers are just replaced by upsampling ones.

Ultimately we would like to use very large and deep convolutional nets on video sequences where the temporal structure provides very helpful information that is missing or far less obvious in static images.

Unsupervised transfer learning may attract more and more attention in the future.

1. <https://keras.io/applications/>

# REFERENCES

1. [https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-](https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53) [the-eli5-way-3bd2b1164a53](https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53)
2. <https://towardsdatascience.com/transfer-learning-from-pre-trained-models-f2393f124751>
3. [https://simpleitk.org](https://simpleitk.org/) [5]<https://neurohive.io/en/popular-networks/resnet/> [6]<https://scikit-learn.org/stable/modules/svm.html> [7]<http://builtin.com/data-science/transfer-learning> [8]<https://openreview.net/forum?id=BJIRs34Fvr>
4. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6640210>
5. <https://arxiv.org/pdf/1409.1556.pdf>
6. <https://www.cse.ust.hk/~qyang/Docs/2009/tkde_transfer_learning.pdf>
7. <https://arxiv.org/pdf/1409.4842.pdf>
8. [https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-](https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf) [networks.pdf](https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf)
9. <https://arxiv.org/pdf/1512.03385.pdf>