Brain Tumor Detection Using Deep-Learning Framework

Submitted in partial fulfillment of the requirements for
the award of
Bachelor of Engineering degree in Computer Science and
Engineering

by

J HARSHINI (Reg. No.

37110299) K GAYATRI (Reg. No.

37110303)

DEPARTMENT OF COMPUTER SCIENCE AND

ENGINEERING SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade “A” by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI
SALAI, CHENNAI – 600 119

MARCH - 2021

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with "A" grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119
www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this project report is the Bonafide work of **JUJJAVARAPU HARSHINI (Reg. No. 37110299)** and **KADAMUTHURU GAYATRI (Reg. No. 37110303)** who carried out the project entitled "**Brain Tumor Detection Using Deeop-Learning Framework**" under my supervision from **August 2020** to **March 2021**.

Internal Guide
Dr. P. Asha, M.E.,Ph.D.,

Head of the Department
Dr. L. Lakshmanan, M.E., Ph.D.,
Dr. S. Vigneshwari, M.E., Ph.D.,

Submitted for Viva voce Examination held on______________________________

Internal Examiner **External Examiner**

DECLARATION

I **JUJJAVARAPU HARSHINI** here by declare that the Project Report entitled "**Brain Tumor Detection Using Deep-Learning Framework"** is done by me under the guidance of **Dr.P.ASHA, M.E.,Ph.d.,** Department of Computer Science and Engineering at Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering**.**

DATE:

PLACE: CHENNAI **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala, M.E., Ph.D., Dean**, School of Computing, **Dr. S. Vigneswari, M.E., Ph.D., and Dr. L. Lakshmanan, M.E., Ph.D., Heads of the Department** of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **D.P.Asha, M.E.,Ph.d.,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Using magnetic resonance imaging to manually segment brain tumors for cancer diagnosis is a complex, tedious and time-consuming task. The accuracy and the robustness of brain tumor segmentation, therefore, are crucial for the diagnosis, treatment planning, and treatment outcome evaluation. Most automated brain tumor segmentation techniques use manually developed functions. Traditional deep learning methods (such as convolutional neural networks) also require a large amount of annotated data for training, which is usually difficult to obtain in the medical field. Here, we describe a new model two-pathway-group CNN architecture for brain tumor segmentation, which exploits local features and global contextual features simultaneously. The model uses the equivalence of the bidirectional CNN model to reduce instability and overfit common parameters. Finally, we merge the cascaded architecture into a two-way multicast CNN, where the output of the basic CNN is processed as an auxiliary source and summarized at the final level. Validation of the models in the data sets BRATS2013 and BRATS2015 shows that the integration of this group CNN into a pathway architecture improved the overall performance over the currently published state-of-the-art while computational complexity remains attractive.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTI

ON

1.1 INTRODUCTION

Tumors will have a huge effect on the brain. Brain cells are destroyed in the area affected by tumor gets and can cause brain collapse. The result of the tumor depends on the size and area affected in the brain. The Brain connected each and every part of the body together to make it perfect sense. If anything happens to the brain our whole system collapses. Some neurons in brain do not have capability to regenerate and there some neurons which stops regeneration as a person age. If the tumor is situated in any of that non- regenerative areas, a person might even lose one of his/her senses. Discovering the tumor at an early stage can save a person's life. Artificial Intelligence is revolutionizing Healthcare in many areas such as Disease Diagnosis with medical imaging, Surgical Robot, maximizing hospital efficiency. Deep learning has been proven to be superior in detecting diseases from X-rays, MRI scans and CT scans which could significantly improve the speed and accuracy of diagnosis. Tumors are located and diagnosed through a very keen medical procedure. Magnetic Resonance Image (MRI) is one such process. We are going to train and validate our model on MRI. These images are sent into to model to train it to detect and locate brain tumor.

Segmenting brain tumors in multi-modal imaging data is a challenging problem due to unpredictable shapes and sizes of tumors. Deep Neural Networks (DNNs) have already been applied to segmentation problems and have shown significant performance improvement compared to the previous methods [4]. We use Convolutional Neural Networks (CNNs) to perform the brain tumor segmentation task on the large dataset of brain tumor MR scans provided by BRATS2015. CNNs are DNNs in which trainable filters and local neighborhood pooling operations are applied alternatingly on the raw input images, resulting in a hierarchy of increasingly complex features. Specifically, we used multi-modality information from T1, T1c, T2 and Flair images as inputs to different CNNs. The multiple intermediate layers apply convolution, pooling, normalization, and other operations to capture the highly nonlinear mappings between inputs and outputs. We take the output of the last hidden layer of each CNN as the

representation of a pixel in that modality and concatenate the representations of all the modalities as features to train a random forest classifier.

Magnetic resonance imaging (MRI) is widely used medical technology for diagnosis of various tissue abnormalities, detection of tumors. The active development in the computerized medical image segmentation has played a vital role in scientific research. This helps the doctors to take necessary treatment in an easy manner with fast decision making. Brain tumor segmentation is a hot point in the research field of Information technology with biomedical engineering. The brain tumor segmentation is motivated by assessing tumor growth, treatment responses, computer-based surgery, treatment of radiation therapy, and developing tumor growth models. Therefore, computer-aided diagnostic system is meaningful in medical treatments to reducing the workload of doctors and giving the accurate results. This chapter explains the causes, awareness of brain tumor segmentation and its classification, MRI scanning process and its operation, brain tumor classifications, and different segmentation methodologies.

CHAPTER 2

LITRECTURE

SURVEY

Habib [1], used artificial convolutional neural network (ANN) to detect tumor with a similar brain tumor dataset employed in this paper. He achieved 88.7% accuracy while testing. He used different neural network which provided him better accuracy. The neural network consists of two max pooling layers and one convolutional 2d (Convo2d) layer in sequential pattern.

Lin and Chang [2], used K-means clustering algorithms with color-based segmentation to track objects of the brain tumor. K-means clustering clusters the similar places together with color. The interesting part in this paper is, they clustered color-spaced image from greyscale with K-means algorithm.

The researchers in [4] used MRI or second resonance images to detect brain tumors. They classified MRI images which is complex because of the variation in brain tumor size and shapes. Decision Tree classifier and Multi-Layer perceptron are the two supervised learning techniques to detect brain tumor.

In this paper [7], the authors explained how misdiagnosis made by image processing or machine learning affect us and showed that they never always give accurate solution or result. There is other variable that need to be taken care while detecting brain tumor. In this paper, they used MRI augmentation technique. Which is sending images in model from various angles and different perspectives. This technique allows model to train on various new images and that got a good result and scores. They used a CNN along with Link Net architecture.

Sharma and Komal [7] proposed a method that involves features detection of brain tumor and classified based on MRI data. They used various filters and image segmentation over images.

Sinthia and Malathi [8] proposed a CNN that automated detection and segmentation of brain tumor. Neural network was constructed using TensorFlow library. BRATS2015 dataset is used by them.

Guotai Wang [9], these researchers proposed an application of deep learning that can interact with user. The application consists of different convolutional neural network 4 architectures that can create a model and can make segmentation of MRI images and can even highlight certain brain oragans. The interesting part is that user can tune and make these organs to pop up. They provided multiple architectures and compared them to get perfect result.

In this paper the authors [10], proposed the essential task carried out by MRI images in diseases diagnosis. They even shared few methods on MRI image processing to implement them in other papers.

Google Net architecture codenamed Inception-v1 is the improved utilization of computing resources inside the network [14]. The network with the inception architecture is faster than the network with non- inception architecture. The Google Net architecture including the inception module uses rectified linear activation function, average pooling layer and not fully connected layer and dropout after removing fully connected layer.

Alex Net [15] architecture is deeper and much greater than Lent architecture. It consists of eight layers, five convolutional layers most of them are followed by max pooling and three fully connected layers. The output is the 1000-way SoftMax that represents the classes. It is trained on two parallel GTX 580 GPU 3 GB which communicate only in certain layers. This scheme reduces the top-5 error rates. Alex Net is improved with Zeit architecture which visualizes the Alex Net activities within the layers to debug problems and obtain better results. It allows observing the evolution of features during training and maps the activities back to the pixel space in intermediate layers.

CHAPTER 3

Aim and Scope

3.1 Aim:

To detected the Brain Tumor using Two-Pathway-Group Conventional Neural Networks. Use the traditional two-way cluster neural network for brain tumor detection.

3.2 Synopsis:

Brain tumor identification is really challenging task in early stages of life. But now it became advanced with deep-learning. Now a day's issue of brain tumor automatic identification is of great interest. In Order to detect the brain tumor of a patient we consider the data of patients like MRI images of a patient's brain. Here our problem is to identify whether tumor is present in patients' brain or not. It is very important to detect the tumors at starting level for a healthy life of a patient. There are many literatures on detecting these kinds of brain tumors and improving the detection accuracies. In this paper, we estimate the brain tumor severity using Convolutional Neural Network algorithm which gives us accurate results.

Chapter 4

SYSTEM DESIGN & METHODOLOGY

4.1 EXISTING SYSTEM

Existing systems describe the automation of cell segmentation. The technique is used to interactive multi label segmentation for N dimensional images. It segments the areas which are more difficult to segment. This method is iterative and provides feedback to the user as the segment is calculated.

4.2 PROPOSED SYSTEM

We take a second dimension to propose a new strategy for MRI of the patient's brain. Here preprocessing is done with Gaussian, which can be a line filter. Then, by identifying the area of the tumor, the GLCM functions are used to extract functions from the image. CNN architecture of bidirectional clusters for brain tumor segmentation, At the same time, it uses its own functions and international information dissemination functions. This model provides equivalence in the two-channel CNN model to reduce backward jitter and overcome parameter sharing. Finally, we integrated the cascaded architecture into the dual-channel CNN pool, during which the basic CNN pins were processed as additional sources, and finally, combined the CNN into a two-way architecture, thereby improving the overall performance. Compared with the currently disclosed progressive method, this method is improved, and the complexity of the process is still quite pleasant.

4.3 REQUIREMENT SPECIFICATIONS

CANCEROUS brain tumors present themselves as unnatural, uncontrolled growth and division of cells in the brain. Although brain tumors are not very common, they are one of the deadliest cancers. For example, in the United States alone, approximately 23,000 new cases of brain cancer were diagnosed in 2015. It is an abnormality in the brain tissues that damage the nervous system severely, which result patient death. are the most common brain tumors that are infiltrative in nature, and occur near white matter fibers. They may spread to any part of the brain making it difficult to detect.

They are considered

high-grade gliomas.
one of the most aggressive tumors with a median survival of 15 months. Glioma can be measured by MRI using a variety of sequences, such as: B. T2-weighted reversal recovery (transition) with fluid attenuation, T1-weighted(T1), contrast-enhanced T1 (T1c) and T2-weighted (T2), use existing automated techniques to segment brain tumors. Healthy brains consist of three types of tissues: gray matter, white matter and cerebrospinal fluid. Detection and segmentation of cancerous cells using MRI not only helps to detect the presence of tumors and their location, but it also enables the identification of tumor size, necrotic tissue, tumorous tissue (vascularized or not) and edema (swelling near the tumor). Brain tumors vary in shape and appearance (gliomas look the same as gliomas, strokes, etc.), which makes them difficult to separate radiologists. Furthermore, they may appear at any location in the brain: depending on the origin of the brain tumor, they can be classified as either primary
tumors or metastatic brain tumors. The edges of brain tumors are often ambiguous and fuzzy, and are hard to distinguish from healthy tissues. Therefore, a more sensitive alternative to MRI is needed to improve the detection of tumors and to increase the survival rate of people with brain

tumors. Machine-assisted image segmentation and subsequent quantification of cancer tissue provide valuable information for early diagnosis and characterization of neuropathology, which can then be used for appropriate treatment. Anatomical structure. In addition, this is very important for early diagnosis, and it helps early prevention by formulating treatment strategies. Cancerous cells are normally quantified by means of the number of lesions, their volume, and biomarkers that have been shown to be related to cognitive deficits. As a result, the quantitative analysis of effected regions requires accurate lesion segmentation, this is challenging due to differences in the size, shape, location and frequency of cancer lesions. Perhaps the most accurate brain tumor segmentation results are achieved manually by an expert; however, this is an expensive, time-consuming, tedious, An impractical task, error-prone, and dependent on differences between observers. As a result, doctors usually only use qualitative or visual inspections, or at best only rough measurements, such as the approximate volume and number of tumors. Manual segmentation of brain tumors from large MRI images is a difficult and time-consuming task. The existing methods of segmenting brain tumors can usually be divided into generative models or discriminatory models [1]. Generative models require prior information and segmentation of brain tumors, whereas discriminative models depend on a set of features and classifiers. The most commonly adopted classifiers are support vector machines (SVMs), random forests, neural
genetic algorithm and network. In contrast, automatic brain tumor segmentation methods use hand-designed features and a variety of image features (e.g., shape, area, perimeter, circularity etc.), intensity (e.g., mean, variance, standard deviations) and texture (e.g., contrast, entropy, correlation etc.).
Recently, deep learning, and in particular the Convolutional Neural Network

(CNN) has become the methodology of choice for medical image analysis, following its tremendous success in routine computer vision applications [2], [3]. Regarding the detection of tumors, the generation of candidates and the reduction of false positives, using deep learning-based methods, unambiguously outperformed traditional machine learning approaches [4]– [7]. This achievement was acknowledged in 2015 where the Deep Medic software for brain lesion segmentation, which was based on a 3D-CNN coupled with a 3D fully connected CRF, won the ISLES 2015 challenge [8]. Additional deep learning-based brain tumor segmentation methods were presented in the 2013, 2015 and 2017 challenges. Different deep learning models were adopted, including FCNN [9]– [4],

[11], 3DCNN [12],

[13], FCNN with CRF [14], 3D U-Net [15], [16] and Autoencoders [17], [18].

4.4 HARDWARE AND SOFTWARE SPECIFICATION

4.4.1 HARDWARE REQUIREMENTS

- Hard Disk : 500GB and Above
- RAM : 4GB and Above
- Processor : I3 and Above

4.4.2 SOFTWARE REQUIREMENTS

- ✔ Operating System : Windows 7, 8, 10 (64 bit)
- ✔ Software : Python
- ✔ Tools : Anaconda (Jupyter Note Book IDE)

4.5 SOFTWARE DESCRIPTION:

PYTHON

• Python is a free, open-source programming language. Therefore, all you have to do is install Python once, and you can start working with it. Not to mention that you can contribute your own code to the community. Python is also a cross-platform compatible language. So, what does this mean? Well, you can install and run Python on several operating systems.

• Python is also a great visualization tool. It provides libraries such as Matplotlib, seaborn and bokeh to create stunning visualizations.

B. PANDAS

• Pandas is a popular Python package for data science, and with good reason: it offers powerful, expressive and flexible data structures that make data manipulation and analysis easy, among many other things. The Data Frame is one of these structures. Pandas is a highlevel data manipulation tool developed by Wes McKinney. It is built on the Numpy package and its key data structure is called the Data Frame. Data Frames allow you to store and manipulate tabular data in rows of observations and columns of variables.

- Pandas is built on top of the NumPy package, meaning a lot of the structure of NumPy is used or replicated in Pandas. Data in pandas is often used to feed statistical analysis in SciPy, plotting functions from Matplotlib, and machine learning algorithms in Scikit-learn. There are two types of data structures in pandas: Series and Data Frames.

a) Series: A pandas Series is a one-dimensional data structure

b) Data Frame: A pandas Data Frame is a two (or more) dimensional data structure basically a table with rows and columns.

4.6 METHODOLOGY

4.6.1 CONVOLUTIONAL NEURAL NETWORK

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are handengineered, with enough training, ConvNets have the ability to learn these filters/characteristics. The architecture of a ConvNet is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlaps to cd over the entire visual area. The objective of the Convolution Operation is to extract the high-level features such as edges, from the input image. ConvNets need not be limited to only one Convolutional Layer. Conventionally, the first Cavalier is responsible for capturing the LowLevel features such as edges, color, gradient orientation, etc. With added layers, the architecture adapts to the High-Level features as well, giving us a network, which has the wholesome understanding of images in the dataset, similar to how we would.

4.6.2 MAGNETIC RESONANCE IMAGING (MRI)

The MRI is a diagnostic tool used for analyzing and studying the human anatomy. The medical images acquired in various bands of the electromagnetic spectrum. The wide variety of sensors used for the acquisition of images and the physics behind them, make each modality suitable for a specific purpose. In MRI, the pictures are produced using a magnetic field, which is approximately 10,000 times stronger than the earth's magnetic field. The MRI produces more detailed images than other techniques, such as CT or ultrasound. The MRI also provides maps of anatomical structures with a high soft-tissue contrast. Basically, the magnetic resonance of hydrogen (1H) nuclei in water and lipid is measured by an MRI

scanner. As the signal values are 12-bit coded, 4096 shades can be represented by a pixel [11]. The MRI scanners require a magnetic field and it is available at 1.5 or 3 T. In comparison with the earth's magnetic field (~50 ft.) the magnetic field of a 3 T MRI scanner is approximately 60,000 times the earth field. The patient is placed in a strong magnetic field, which causes the protons in the water molecules of the body to align either in a parallel or anti-parallel orientation with the magnetic field. A radiofrequency pulse is introduced, causing the spinning protons to move out of the alignment. When the pulse is stopped, the protons realign and emit radio frequency energy signal that is localized by the magnetic fields and are spatially varied and rapidly turned on and off. A radio antenna within the scanner detects the signal and creates the image. Functional MRI is a technique for examining the brain activation, which unlike PET, is non-invasive with relatively high spatial resolution.

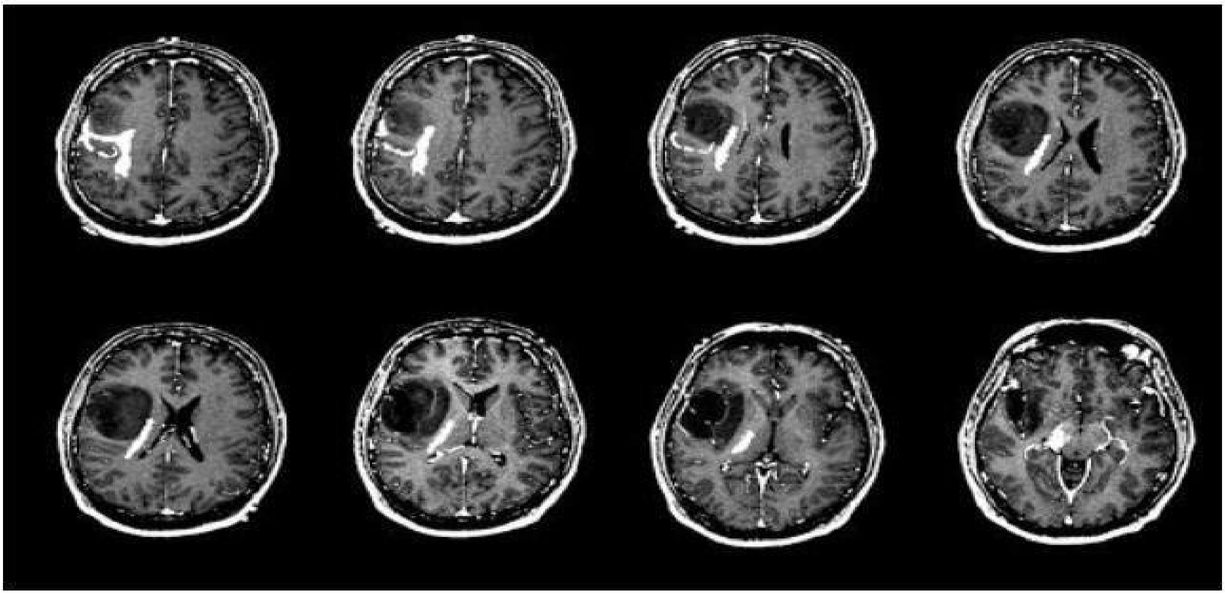


Figure 4.6 MRI of Human Brain

The most common method utilizes a technique called blood oxygen level dependent contrast. This is an example of endogenous contrast, making use of the inherent signal differences in blood oxygenation content. In the normal resting state, a high concentration of deoxyhemoglobin attenuates the MRI signal due to its paramagnetic nature. However, the neuronal activity, in response to some task or stimulus, creates a local demand for the oxygen supply, which increases the fraction of oxy hemoglobin causing a signal increase on T2 or T2*-weighted images. In a typical experiment, the patient is subjected to a series of rest and task intervals, during which MRI images are repeatedly acquired. A radio antenna within the scanner detects the signal and creates the image. Functional MRI is a technique for examining the brain activation, which unlike PET, is non-invasive with relatively high spatial resolution. The signal changes during the course of time are then examined on a pixelby-pixel basis to test how well they correlate with the known stimulus pattern. The pixels that demonstrate a statistically significant correlation are highlighted in color and overlaid onto a grayscale MRI image to create an activation map of the brain. The location and extent of activation is linked to the type of stimulus. Thus, a simple thumbfinger movement task will produce activation in the primary motor cortex.

Chapter 5

DESIGN AND IMPLEMENTATION

Constraints in Analysis

- Constraints as Informal Text
- Constraints as Operational Restrictions
- Constraints Integrated in Existing Model Concepts
- Constraints as a Separate Concept
- Constraints Implied by the Model Structure

Constraints in Design

- Determination of the Involved Classes
- Determination of the Involved Objects
- Determination of the Involved Actions
- Determination of the Require Clauses
- Global actions and Constraint Realization

Constraints in Implementation

The hierarchy of relationships can lead to more classes and more complex implementation structures. Therefore, it is recommended to convert the hierarchical structure of the relationship to a simpler structure, such as a classic plane structure. The developed hierarchical model can be easily converted into a flat two-part model. On the one hand it is composed of classes, on the other it is composed of flat relationships. Since flat relationships are easy to implement, they are preferred at the design level. There are no logos or functions related to the plane relationship. The plane relationship follows the concept of entity relationship modeling and many object-oriented technologies.

5.1 Nonfunctional Requirements

Performance Requirement: The application at this side controls and communicates with the following three main general components. embedded browser in charge of the navigation and accessing to the web service.

Server Tier: The server side contains the main parts of the functionality of the proposed architecture. The components at this tier are the following.

Web Server, Security Module, Server-Side Capturing Engine, Preprocessing Engine, Database System, Verification Engine, Output Module.

Safety Requirements

1.Software is essential for safety. In this case, there are issues related to the integrity level.

2. Even if the software is part of a security-critical system, it may not necessarily be critical to security. For example, a program can only write transactions.

3. If the system is to have a higher integrity level, and it is shown that the software has this integrity level, the hardware must have at least the same integrity level.

4. If the hardware and software of the system are (in a broad sense) unreliable, it makes no sense to write "perfect" code in any language.

5. If a computer system is to run software with a higher integrity level, the system should not support software with a lower integrity level at the same time.

6.Systems with different security level requirements should be separated.

7.Otherwise, the highest level of integrity should be applied to all systems in the same environment.

Architecture Diagram:

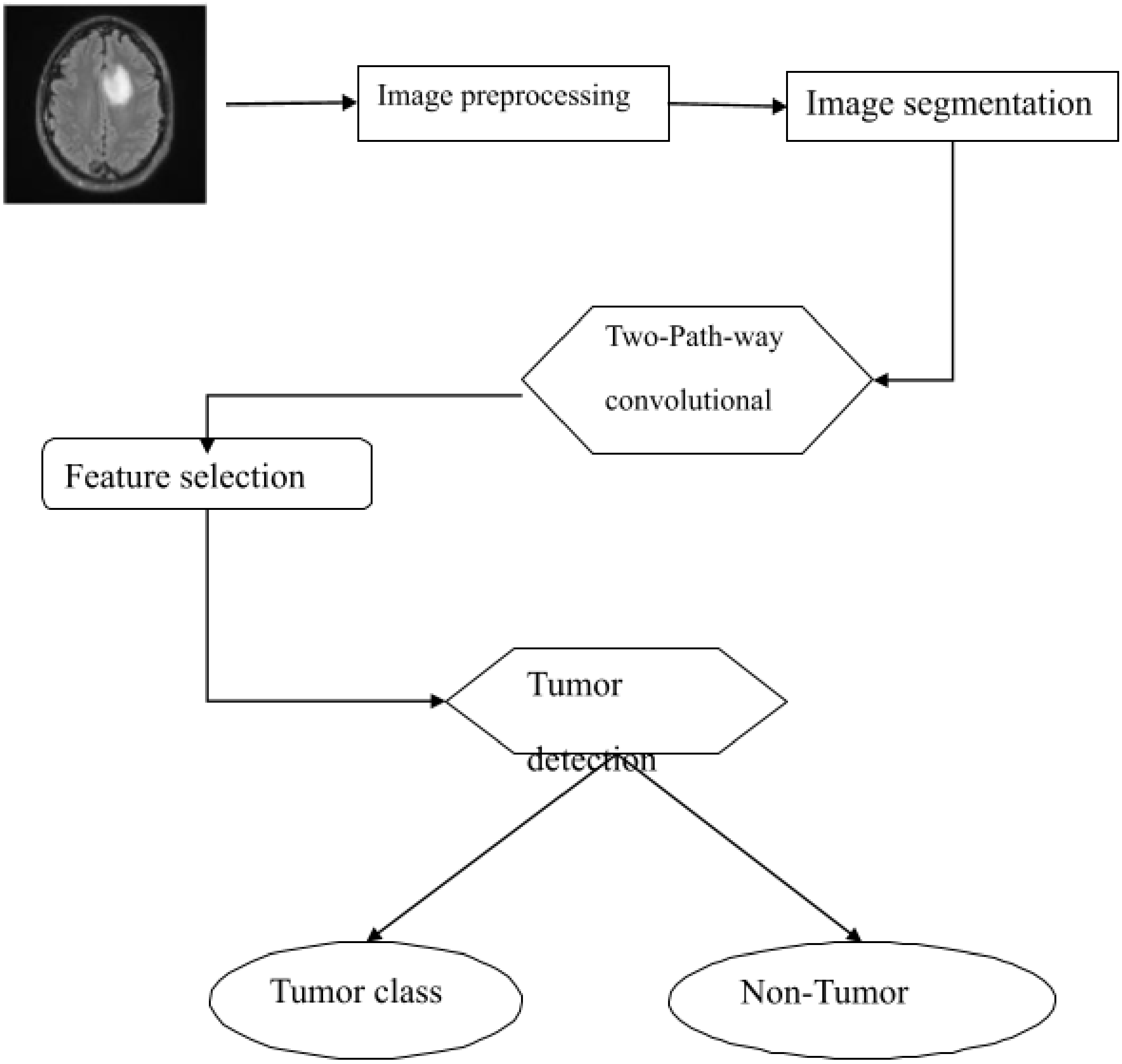


Fig: 5.1

5.2 Sequence Diagram:

A **s**equence diagram is an interaction diagram that shows how and in what order the processes interact**.** This is the construction of message sequence diagrams**,** sometimes called event diagrams, event scenarios**,** and sequence diagram**s**.

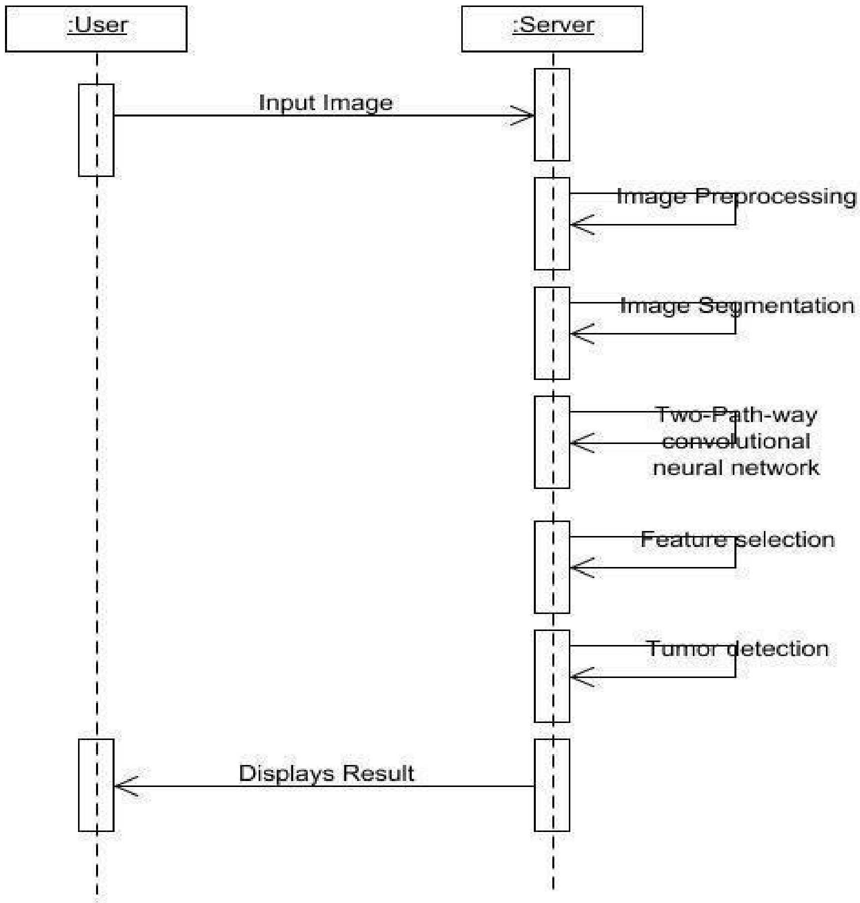


Fig: 5.2

5.3 Use Case Diagram:

Unified Modeling Language (UML) is a universal, standardized modeling language used for software development. This standard is managed and created by the asset management team. UML contains many graphical markup techniques for creating visual models of software-intensive systems. It is used to define, visualize, modify, construct and record the artifacts of object-oriented, software-intensive systems under development. Unified Modeling Language (UML) is a universal modeling language that has been standardized in the field of software development. This standard is managed and created by the asset management team. It is used to define, visualize, modify, construct and document the artifacts of intensively developed object-oriented software systems.

Use case Diagram

Use case diagrams are used to graphically describe the functions provided by the system based on participants, their goals, and any dependencies between these use cases. The use case diagram consists of two parts:

Use case: A use case describes a series of actions through which the subject can be measured and drawn as a horizontal ellipse.

Participants: Participants are individuals, organizations, or external systems that play a role in one or more interactions with the system.

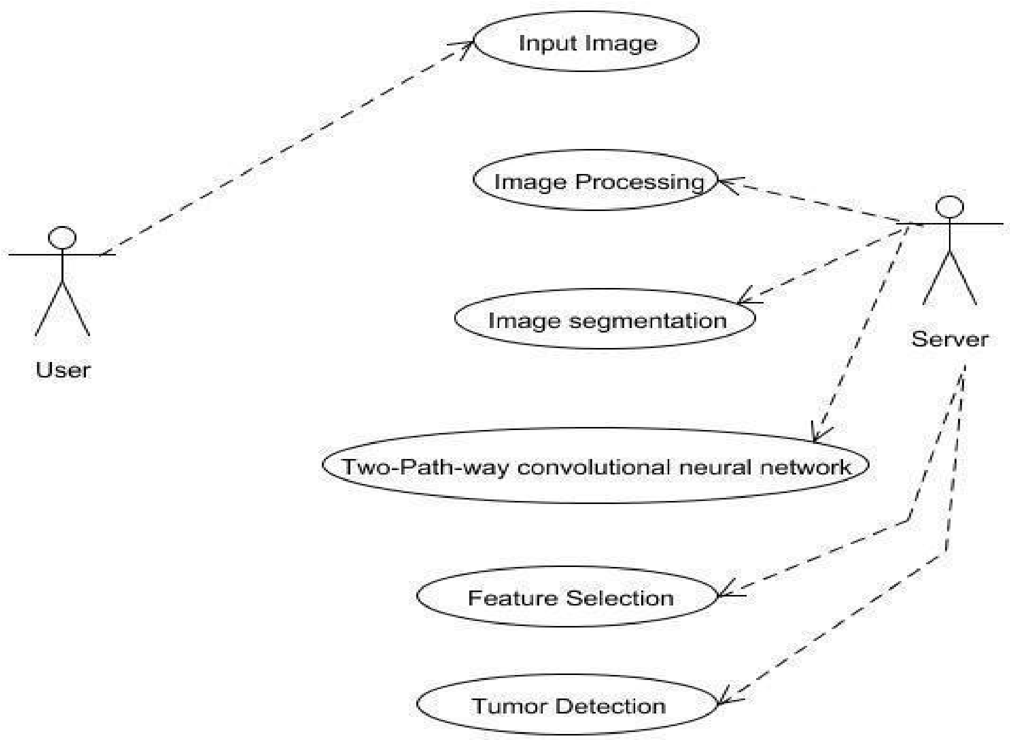


Fig: 5.3

5.4 Activity Diagram:

Activity diagram is a graphical representation of workflows of stepwise activities and actions with support for choice, iteration and concurrency. An activity diagram shows the overall flow of control.

The most important shape types:

- Rounded rectangles represent activities.
- Diamonds represent decisions.
- Bars represent the start or end of concurrent activities.
- A black circle represents the start of the workflow.
- An encircled circle represents the end of the workflow.

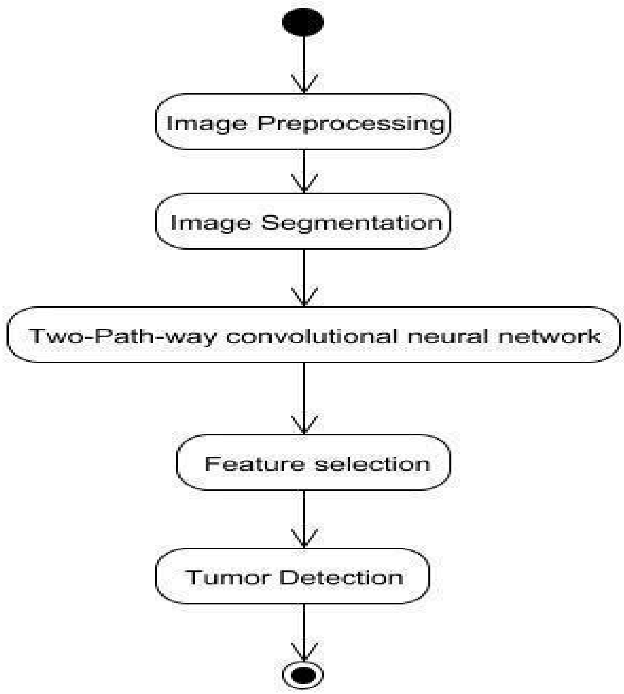


Fig: 5.4

5.5 Collaboration Diagram:

UML collaboration diagram illustrates the relationship and interaction between software objects. They assume that use cases, system usage contracts and domain models already exist. The collaboration diagram shows the messages sent between the class and the object.

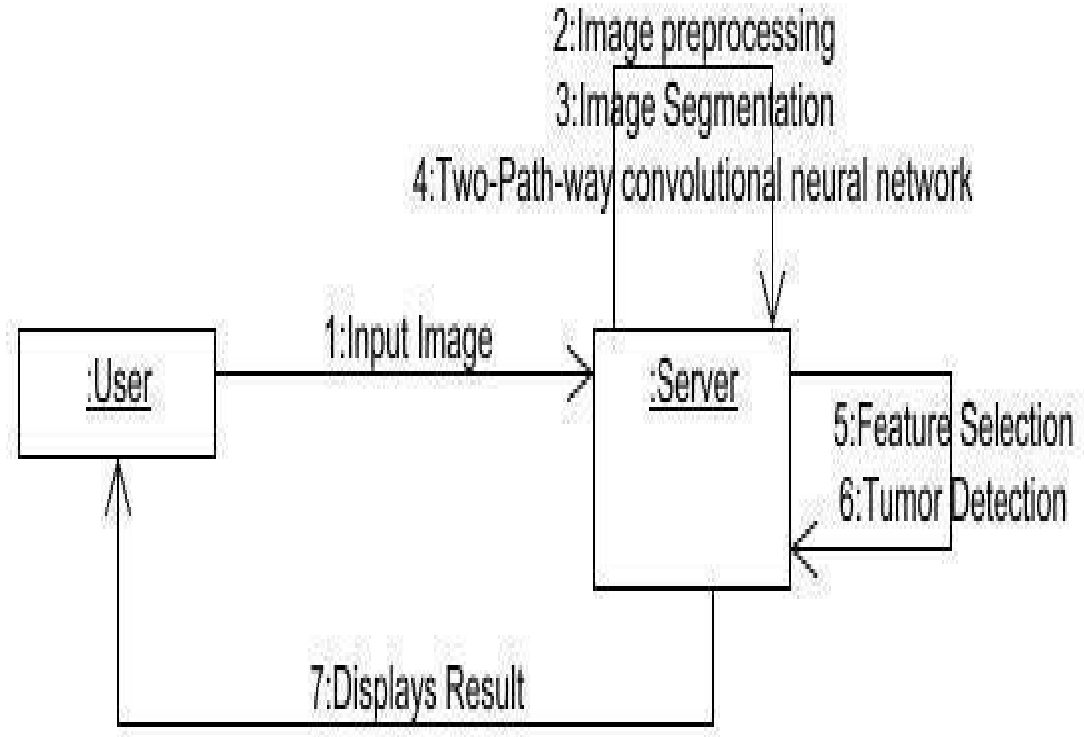


Fig: 5.5

5.6 MODULES

- Image acquisition
- Image preprocessing
- Image segmentation
- Convolutional neural network
- Tumor detection

MODULE EXPLANATION:

5.6.1 Image Acquisition

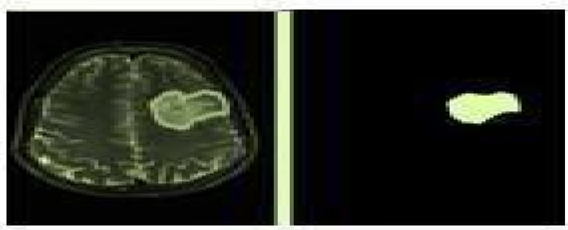
The Primary Phase is acquiring images. After the Images collection, the obtained images have to be prepared with a wide range of vision. First capture the input images from available source

5.6.2 Pre-Processing

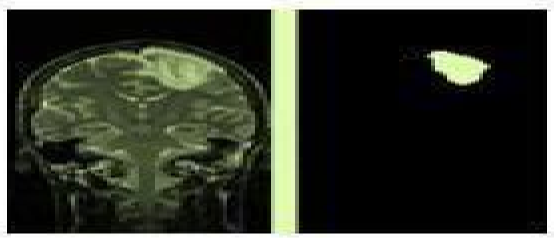
The images which are collected are subjected to pre- processing. In Pre- processing stage basic steps are image resizing and applying Gaussian filters for a perfect input clear image for easy identification of an image. Pre-processed images will be segmented digitally into various pixels. We do this segmentation for an image is to modify its representation to have more clarity to analyze the images.

5.6.3 image segmentation

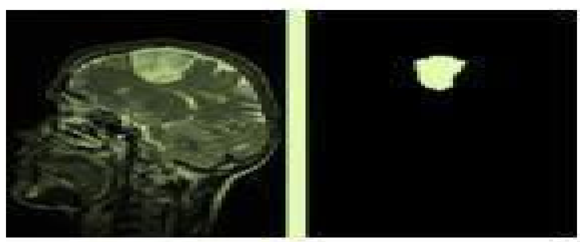
In the first stage, the pre-processed brain Magnetic Resonance image will be transformed into a binary image with a threshold of 128 for the cutoff. Pixel values higher than the specified thresholds are mapped as white, with other regions marked as black; these two allow various regions to be generated around the disease. In the second stage, an erosion process of morphology is used to extract white pixels. Eventually, the eroded area and the original image are separated into two equal areas, and the region with black pixels from the eroding is counted as a mask of brain Magnetic Resonance image. In this paper, wavelet transformation is used for the efficient segmentation of the brain Magnetic Resonance image. Figure 3 shows the fully automatic heterogeneous segmentation. Figure 3(a) shows the axial image and its segmentation figure 3(b) Coronal image and its segmentation figure 3 (c) Sagittal images and its segmentation.



3(a)



3(b)



3(c)

***Feature* Extraction**

In the feature extraction process, we can implement the effective texture operator which labels the pixels of an image. Here we extract the features and characteristics of Images for easy detection of brain tumor.

Classification

Convolutional neural networks algorithm is used for classification of brain images. It is producing the best results for the image

Tumor Detection

Finally, analyze the image using filters and Convolutional neural networks algorithm to detect the tumor or Non-tumor.

5.7 TEST PROCEDURE

System Testing:

Testing is performed to identify errors. It is used for quality assurance. Testing is an integral part of the entire development and maintenance process. The goal of the testing during phase is to verify that the specification has been accurately and completely incorporated into the design, as well as to ensure the correctness of the design itself. For example, the design must not have any logic faults in the design is detected before coding commences, otherwise the cost of fixing the faults will be considerably higher as reflected. Detection of design faults can be achieved by means of inspection as well as walkthrough.

Testing is one of the important steps in the software development phase. Testing checks for the errors, as a whole of the project testing involves the following test cases:

Static analysis is used to investigate the structural properties of the Source code.

Dynamic testing is used to investigate the behavior of the source

code by executing the program on the test data.

5.7.1. TEST DATA AND

OUTPUT UNIT TESTING

Unit testing is conducted to verify the functional performance of each modular component of the software. Unit testing focuses on the smallest unit of the software design (i.e.), the module. The white-box testing techniques were heavily employed for unit testing.

5.7.2 FUNCTIONAL TESTS

Functional test cases involved exercising the code with nominal input values for which the expected results are known, as well as boundary values and special values, such as logically related inputs, files of identical elements, and empty files.

Three types of tests in Functional test:

Performance Test

Stress Test

Structure Test

PERFORMANCE TEST

It determines the amount of execution time spent in various parts of the unit, program throughput, and response time and device utilization by the program unit.

STRESS TEST

Stress Test is those tests designed to intentionally break the unit. A Great deal can be learned about the strength and limitations of a program by examining the manner in which a programmer in which a program unit breaks.

STRUCTURED TEST

Structure Tests are concerned with exercising the internal logic of a program and traversing particular execution paths. The way in which White-Box test strategy was employed to ensure that the test cases could Guarantee that all independent paths within a module have been have been exercised at least once.

Exercise all logical decisions on their true or false sides.

Execute all loops at their boundaries and within their operational bounds.

Exercise internal data structures to assure their validity.

Checking attributes for their correctness.

Handling end of file condition, I/O errors, buffer problems and textual errors in output information

Integration testing is a systematic technique for construction the program structure while at the same time conducting tests to uncover errors associated with interfacing. i.e., integration testing is the complete testing of the set of modules which makes up the product. The objective is to take untested modules and build a program structure tester should identify critical modules. Critical modules should be tested as early as possible. One approach is to wait until all the units have passed testing, and then combine them and then tested. This approach is evolved from unstructured testing of small programs. Another strategy is to construct the product in increments of tested units. A small set of modules are integrated together and tested, to which another module is added and tested in combination. And so on. The advantages of this approach are that, interface dispenses can be easily found andcorrected.

The major error that was faced during the project is linking error. When all the modules are combined the link is not set properly with all support files. Then we checked out for interconnection and the links. Errors are localized to the new module and its intercommunications. The product development can be staged, and modules integrated in as they complete unit testing.

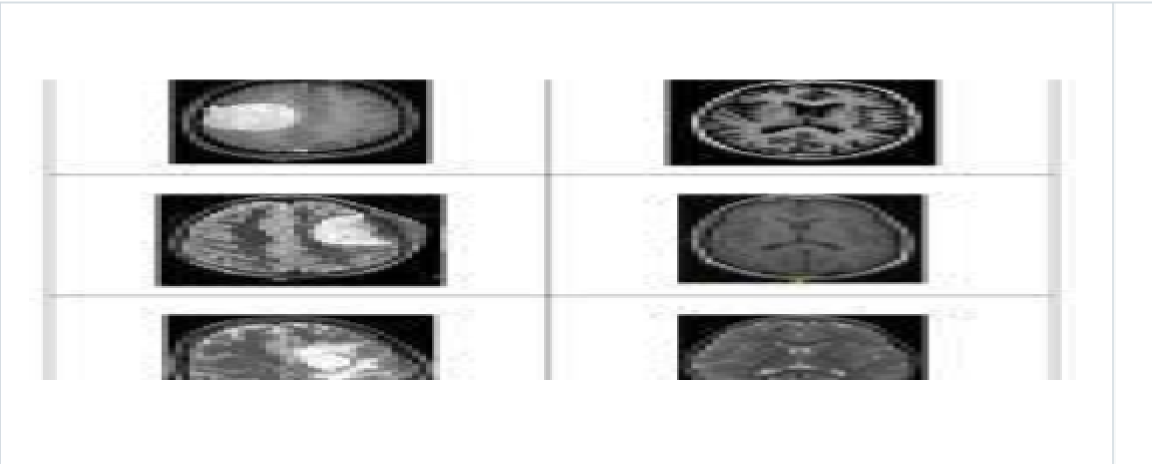
Testing is completed when the last module is integrated and tested.

CHAPTER 6

RESULT AND CONCLUSION

Result:

Our data set contains tumor and non-tumor MRI images obtained from various online sources. Use convolutional neural network for detection. Modeling is done using Python language. Calculate the accuracy and compare it with all other modern methods.



To determine the effectiveness of the proposed brain, training accuracy, verification accuracy, and verification loss need to be calculated. Tumor classification scheme. The current technology for detecting brain tumors uses SVM (Support Vector Machine) classification. Feature extraction requires output. Based on the feature value, the classification output is generated and the accuracy is calculated. Tumor and non-tumor detection based on support vector machines take a long time and have poor calculation accuracy. The proposed CNN-based classification does not require a separate feature extraction step. The value of this function is taken from CNN itself. In the picture. The classification results of tumor and non-tumor brain imaging are shown. Therefore, the complexity and calculation time are low and accurate. The figure shows the results of brain tumor classification accuracy. Finally, according to the value of the probability score, it is classified as brain tumor or non- tumor brain. Normal brain imaging is the least likely. The score value compared with normal and neoplastic brains.

CONCLUSION

Our data set includes tumor MRI images and non-tumor images obtained from various online sources. Radiation podia contains real patient cases. Tumor images are obtained from the test data set of "Radio podia and Brain Tumor Image Segmentation Benchmark (BRATS) 2015". The detection is carried out through a convolutional network. Modeling is done using Python language. Calculate the accuracy and compare it with all other modern methods. In order to determine the effectiveness of the proposed brain, training accuracy, verification accuracy, and verification loss need to be calculated.

Tumor classification scheme. The current technology for detecting brain tumors uses SVM (Support Vector Machine) classification. Feature extraction requires output. Based on the feature value, the classification output is generated and the accuracy is calculated. Tumor and non-tumor detection based on support vector machines take a long time and have poor calculation accuracy. The proposed CNN-based classification does not require a separate feature extraction step. The value of this function is taken from CNN itself. In the picture. The classification results of tumor and non-tumor brain imaging are shown. Therefore, the complexity and calculation time are low and accurate. The figure shows the results of brain tumor classification accuracy. Finally, according to the value of the probability score, it is classified as brain tumor or non-tumor brain. Normal brain imaging is the least likely. The score value compared with normal and neoplastic brains.

SCREEN SHOT

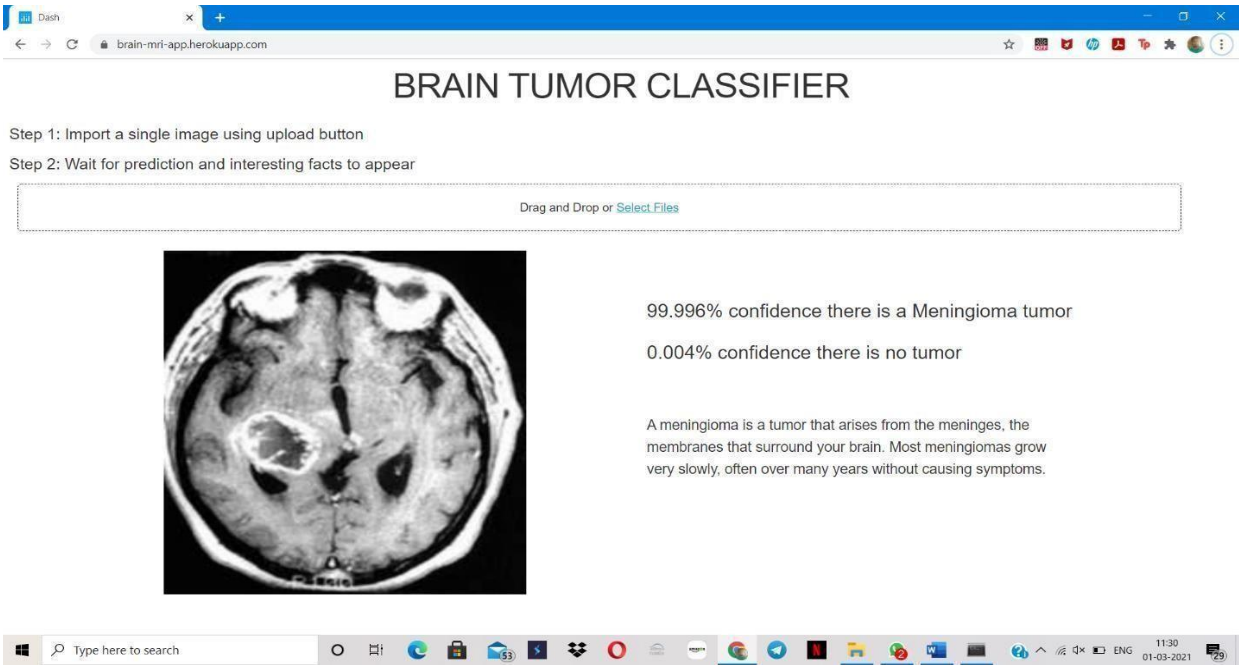


Figure : E.1 Meningioma Tumor classifier

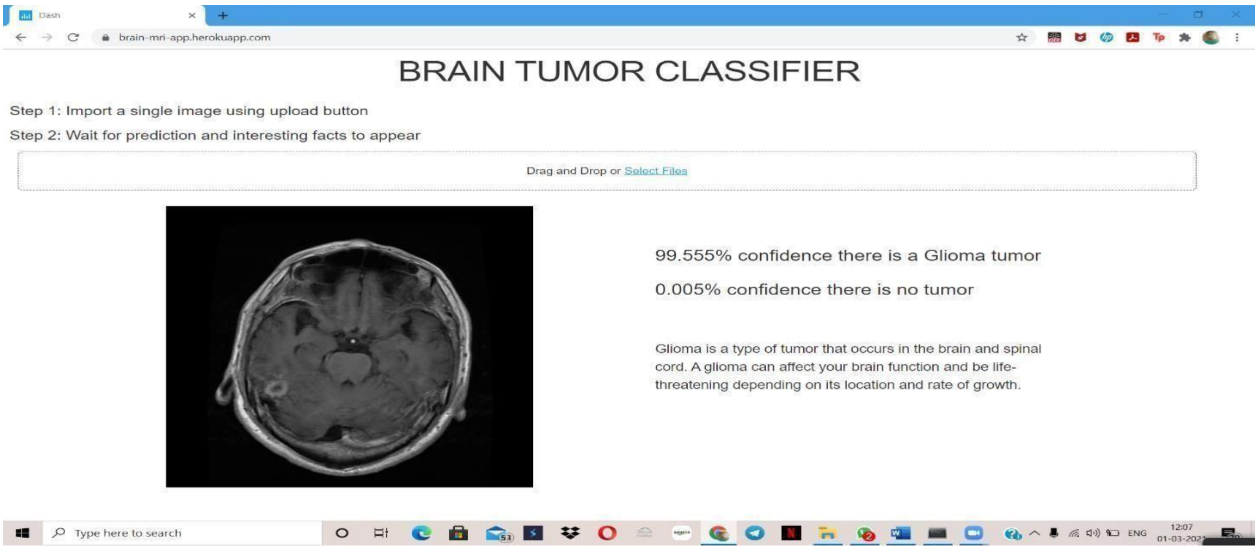


Figure: E.2 Glioma Tumor Classifier

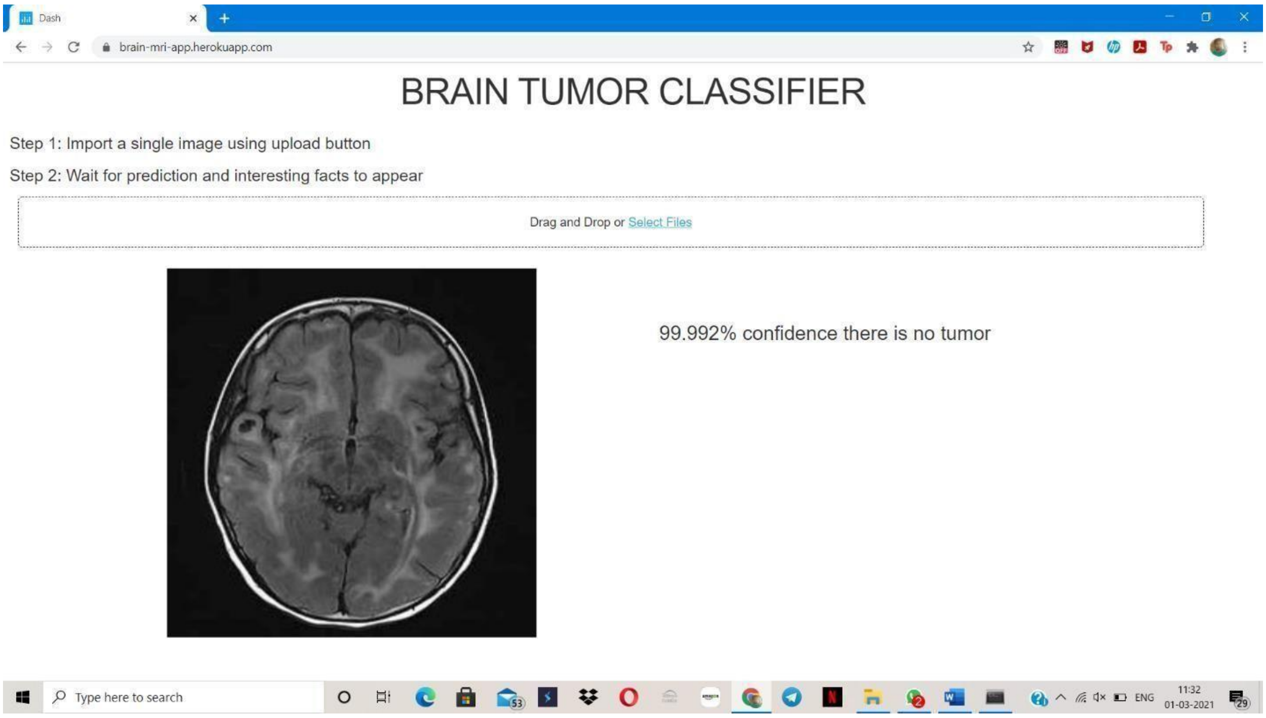


Figure: E3 No Tumor Classifier

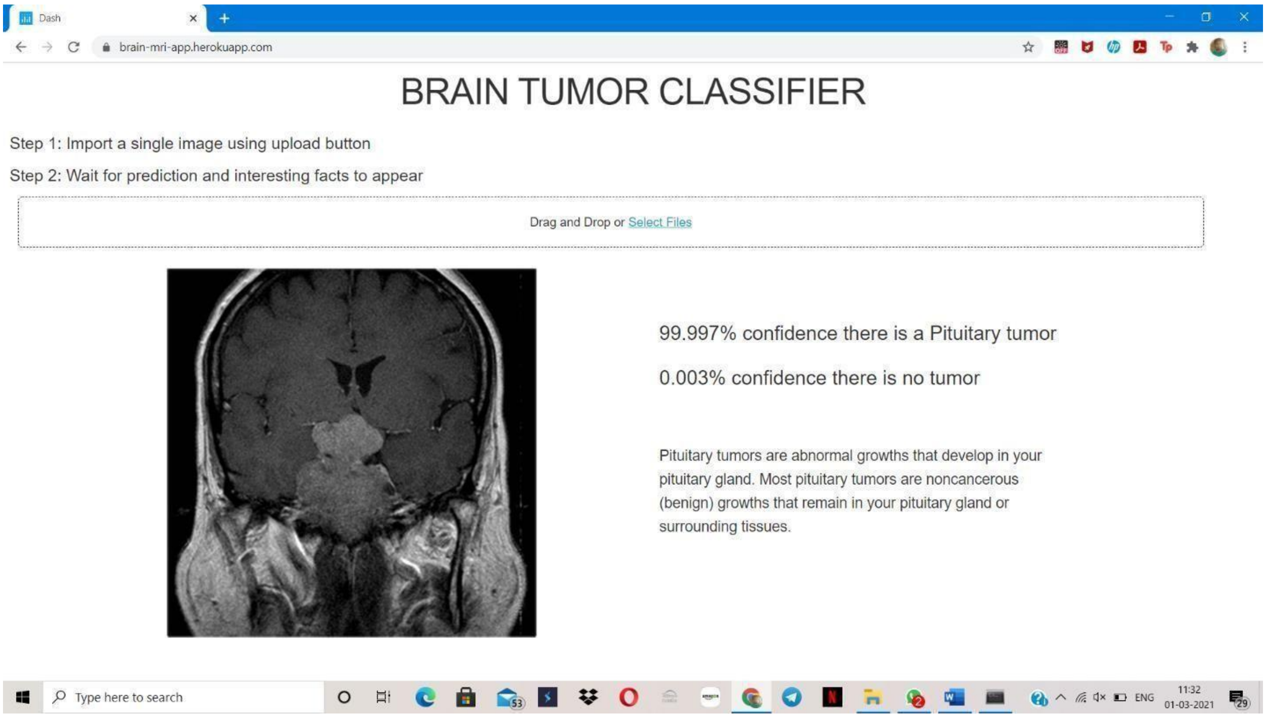


Figure: E4 Pituitary Tumor Classifier

SOURCE CODE

Import libraries

Nc import numpy as
np import pandas as
pd import seaborn
as sns

import matplotlib.pyplot as
plt import cv2

from sklearn.model_selection
import train_test_split

Store every image in an array

```
import os
data_dir = ('D:/Brain-Tumor-Classification/Training/')
categories = ['glioma_tumor', 'meningioma_tumor', 'no_tumor',
'pituitary_tumor'] for i in categories:
    path =
    os.path.join(data_dir, i) for
    img in os.listdir(path):
   img_array = cv2.imread(os.path.join(path,img))
```

Resize each image to same size for fast processing

img_size = 128
image_array = cv2.resize(img_array, (img_size,img_size))

gt1 = cv2.imread('D:/Brain-Tumor-Classification/Training/glioma_tumor/gg(1).jpg')
mt1 =
cv2.imread('D:/Brain-Tumor-Classification/Training/meningioma_tumor/m
(10).jpg')nt1 = cv2.imread('D:/Brain-Tumor-Classification/Training/no_tumor/1.jpg') pt1 =
cv2.imread('D:/Brain-Tumor-Classification/Training/pituitary_tumor/p(151).jp
g')

Example of Glioma Tumor

```
plt.rcParams["figure.figsize"] =
(5,5) plt.imshow(gt1)
```

plt.axis('off')

Example of Meningioma Tumor

plt.rcParams["figure.figsize"] =
(5,5) plt.imshow(mt1)

plt.axis('off')

Example of No Tumor

plt.rcParams["figure.figsize"] =

(5,5) plt.imshow(nt1)

plt.axis('off')

Convert each image to grayscale and append into an array

```
train_data = []


for i in categories:
    train_path =
    os.path.join(data_dir,i) tag =
    categories.index(i)
    for img in
        os.listdir(train_path): try:
            image_arr = cv2.imread(os.path.join(train_path , img),
cv2.IMREAD_GRAYSCALE)
            new_image_array = cv2.resize(image_arr, (img_size,img_size))
            train_data.append([new_image_array , tag])
        except Exception
    as e: pass
```

Import keras' functions to create CNN

modelfrom sklearn.metrics import

confusion_matrix import itertools

from keras.utils.np_utils import to_categorical

```
from keras.models import Sequential
from keras.layers import Dense, Dropout, Flatten, Conv2D,
MaxPool2D from keras.optimizers import RMSprop,Adam
from keras.preprocessing.image import
ImageDataGenerator from keras.callbacks import
ReduceLROnPlateau
from keras.callbacks import

EarlyStopping model = Sequential()

model.add(Conv2D(filters = 64, kernel_size = (5,5),padding = 'Same', activation ='relu',
input_shape = (128,128,1)))
model.add(MaxPool2D(pool_size=(2,2)))
model.add(Dropout(0.2))

model.add(Conv2D(filters = 128, kernel_size = (3,3),padding = 'Same', activation
='relu'))
model.add(MaxPool2D(pool_size=(2,2),
strides=(2,2))) model.add(Dropout(0.2))

model.add(Conv2D(filters = 128, kernel_size = (3,3),padding = 'Same', activation
='relu'))
model.add(MaxPool2D(pool_size=(2,2),
strides=(2,2))) model.add(Dropout(0.2))

model.add(Conv2D(filters = 128, kernel_size = (2,2),padding = 'Same', activation
='relu'))
model.add(MaxPool2D(pool_size=(2,2),
strides=(2,2))) model.add(Dropout(0.2))

model.add(Conv2D(filters = 256, kernel_size = (2,2),padding = 'Same', activation
='relu'))
```

```
model.add(MaxPool2D(pool_size=(2,2),
strides=(2,2))) model.add(Dropout(0.2))

model.add(Flatten())

model.add(Dense(1024, activation =
"relu")) model.add(Dropout(0.5))

model.add(Dense(4, activation = "softmax"))

optimizer = Adam(lr=0.001, beta_1=0.9, beta_2=0.999)

model.compile(optimizer = optimizer , loss = "categorical_crossentropy",
metrics=["accuracy"])

epochs = 20

es = EarlyStopping(
   monitor='val_acc'
   , mode='max',
   patience = 3
)

batch_size = 16
imggen =
      ImageDataGenerator(
      featurewise_center=Fal
      se,
      samplewise_center=False,
      featurewise_std_normalization=Fa
      lse,
      samplewise_std_normalization=Fa
```

lse, zca_whitening=False,

rotation_range=0,

```
zoom_range = 0,
width_shift_range
=0,
height_shift_range
=0,
horizontal_flip=Tru
e,
vertical_flip=False)
```

Fit the model with Train and Validation data sets

```
from keras.callbacks import ModelCheckpoint


checkpoint = ModelCheckpoint("model_weights.h5", monitor='val_acc', verbose=1,
save_best_only=True, mode='max')
callbacks_list = [checkpoint]


imggen.fit(X_train)
history = model.fit_generator(imggen.flow(X_train,y_train,batch_size =
                    batch_size), epochs = epochs, validation_data =
                    (X_val,y_val), steps_per_epoch = X_train.shape[0] //
                    batch_size,
                callbacks=callbacks_
list) serialize model structure to
JSON model_json =
model.to_json()
with open(r"D:\Brain-Tumor-Classification\model.json", "w") as
json_file: json_file.write(model_json
```

REFERENCE

[1]Abhishek Anil, Aditya Raj, H Aravind Sarma, Naveen Chandran R,Deepa P L, "Brain Tumor detection from brain MRI using Deep Learning" International Journal of Innovative Research in Applied Sciences and Engineering (IJIRASE), Volume 3, Issue 2, DOI:10.29027/IJIRASE.v3.i2.2019, 458-465, August 2019.

[2] George, Dena Nadir, Hashem B. Jehlol, and Anwer Subhi Abdulhussein. "Brain Tumour Detection Using Shape features and Machine Learning Algorithms".

[3] Işın, Ali, Cem Direkoğlu, and Melike Şah. "Review of MRI-based brain tumor image segmentation using deep learning methods." Procedia Computer Science 102 (2016): 317-324

[4] Leiner, Tim, et al. "Machine learning in cardiovascular magnetic resonance: basic concepts and applications." Journal of Cardiovascular Magnetic Resonance 21.1 (2019): 61.

[5] Lundervold, Alexander Selvikvåg, and Arvid Lundervold. "An overview of deep learning in medical imaging focusing on MRI." Zeitschrift für

Medizinische

Physik 29.2 (2019): 102-127.

[6] Long, C., Basharat, A., Hoogs, A.: A Coarse-to-fine Deep Convolutional Neural Network Framework for Frame Duplication Detection and Localization

in Forged Videos, p. 10 (2018)

[7] M. Wu, C. Lin and C. Chang, "Brain Tumor Detection Using ColorBased K- Means Clustering Segmentation," Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP 2007), Kaohsiung, 2007.

[8] Mohsen, Heba, et al. "Classification using deep learning neural networks for brain tumors." Future Computing and Informatics Journal 3.1 (2018): 68-71.

[9] Nogovitsyn, N., et al.: Testing a deep convolutional neural network for automated hippocampus segmentation in a longitudinal sample of healthy participants. NeuroImage 197, 589–597 (2019).

[10] Sharma, Komal, Akwinder Kaur, and Shruti Gujral. "Brain tumor detection based on machine learning algorithms." International Journal of Computer Applications 103.1 (2014).

[11] Sobhaninia, Zahra, et al. "Brain tumor segmentation using deep learning by type specific sorting of images." arXiv preprint arXiv:1809.07786 (2018).

[12] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.:

Rethinking the Inception Architecture for Computer Vision,
arXiv:1512.00567 [cs], December
2015

[13]Wang, Guotai, et al. "Interactive medical image segmentation using deep learning with image-specific fine tuning." IEEE transactions on medical imaging
37.7 (2018): 1562-1573.

[14] Zacharaki, E.I., et al.: Classification of brain tumor type and grade using MRI texture and shape in a machine learning scheme. Magn. Reson. Med. 62(6), 1609–1618 (2009). https://doi.org/10.1002/mrm.22147

[15] https://en.wikipedia.org/wiki/Convolutional_neural_network , December, 21 2019

[16] https://en.wikipedia.org/wiki/Loss_function , December, 21 2019

[17] https://en.wikipedia.org/wiki/Mathematical_optimization , December, 21 2019.

GENERAL PAPER

A)

Segmentation of brain tumors victimisation ancient multi-scale bilateral Conventional Neural Networks

J. Harshini
Department of CSE Final Year.
Sathyabama Institute of Science and technology
Chennai, Tamil Nadu
Harshinijajjavarapu@gmail.com

K.Gayatri
Department of CSE Final Year.
Sathyabama Institute of Science and technology
Chennai, Tamil Nadu
gayathriknaidu456@gmail.com

DR.P Asha
Assocaite Professor
Department of CSE Final Year.
Sathyabama Institute of Science and technology
Chennai, Tamil Nadu
asha.cse@sathyabama.ac.in

ABSTRACT

Manual segmentation of brain tumors from MRI imaging to detect cancer is a tedious, tedious, and tedious task. Therefore, the accuracy and reliability of brain tumor segmentation is also key to decision-making, treatment design, and analysis of treatment results. The automated brain tumor segmentation strategy uses manually developed parameters. As with convolutional neural networks, training requires a large amount of annotated data that is generally difficult to obtain in the medical field. The model offers the same variations as the two-channel CNN model to reduce jitter and overfitting parameters. Finally, we want to introduce a two-way cascade design to CNN, where the base CNN output is processed as an additional source and combined at the last level. When testing the model in the BRATS2013 and BRATS2015 knowledge sets, the CNN integrated package in the 2-channel design was not hidden. Compared to the progressive method described above, the overall performance has been improved and the process quality remains attractive.

B)

ORIGINALITY REPORT

8%	1%	8%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Muhammad Imran Razzak, Muhammad Imran, Guandong Xu. "Efficient Brain Tumor Segmentation With Multiscale Two-Pathway-Group Conventional Neural Networks", IEEE Journal of Biomedical and Health Informatics, 2019 Publication	6%
2	"Abstracts of the 4th World Congress on Developmental Origins of Health and Disease (DOHaD), 13-16 September 2006, Educatorium, University of Utrecht, The Netherlands", Early Human Development, 200608 Publication	1%
3	"Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries", Springer Science and Business Media LLC, 2018 Publication	<1%

Exclude quotes Off

Exclude matches Off