

Brain Tumor MRI image Segmentation Using Convolutional Neural Network

A thesis
By

MD. MOSABBIR HOSSAIN
MD. SHAHIDUL ISLAM SHABUZ
KAZI ZIAUL MAMUN



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
DHAKA UNIVERSITY OF ENGINEERING & TECHNOLOGY,
GAZIPUR

JULY 2021

Brain Tumor MRI image Segmentation Using Convolutional Neural Network

A Thesis

by

MD. MOSABBIR HOSSAIN

Student No.: 154005

Reg. No.: 9052, Session:2018-2019

MD. SHAHIDUL ISLAM SHABUZ

Student No.: 154010

Reg. No.: 9057, Session: 2018-2019

KAZI ZIAUL MAMUN

Student No.154015

Reg. No.: 9062, Session: 2018-2019

Supervisor:

Dr. Mohammad Abul Kashem

Professor

Department of Computer Science and Engineering, DUET

Submitted to the

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
DHAKA UNIVERSITY OF ENGINEERING & TECHNOLOGY, GAZIPUR

In partial fulfillment of the requirements for the award of the degree
of

BACHELOR OF SCIENCE IN COMPUTER SCIENCE & ENGINEERING

JULY 2021

DECLARATION

We hereby declare that this submission is our own work and that, to the best of our knowledge, it does not contain any idea, material or resources published or written by another person except otherwise explicitly mentioned and properly referenced. We also declare that neither a part nor a whole of this work has been submitted for an award of any other degree of a university or other institute of higher education.

Signature of authors

.....

(Md Mosabbir Hossain)

.....

(Md Shahidul Islam Shabuz)

.....

(Kazi Ziaul Mamun)

Student of department of Computer Science and Engineering of Dhaka University Engineering and Technology, Gazipur.

Signature of Thesis Supervisor

.....

(Supervisor)

Dr. Mohammad Abul Kashem

Professor

Department of Computer Science and Engineering
Dhaka University of Engineering & Technology, Gazipur

ACKNOWLEDGEMENT

All the praises are due to almighty Allah for giving us the ability to successfully complete this thesis and project. We would like to acknowledge the supports provided by different people in different aspect of our research. First of all we express our sincere gratitude to our thesis supervisor **Dr. Mohammod Abul Kashem**, Professor, Department of Computer Science and Engineering, Gazipur, for providing us time to time advices and instructions in progression of the research. We are also thankful to him insightful review during the editing and writing of this dissertation. We wish to the all the teacher who have directly or indirectly contributed towards the completion of the thesis work. Last but not least, we are grateful to our parents and to our families for their patience and support during our studies.

July, 2021

Authors

Md Mosabbir Hossain
Md Shahidul Islam Shabuz
Kazi Ziaul Mamun

ABSTRACT

In medical image processing, brain tumor segmentation is a critical task. Early detection of brain tumors is crucial for enhancing treatment options and increasing the survival rate of patients. Brain tumor segmentation by manual for cancer diagnosis is a challenging and time-consuming task to extract information from a big number of MRI images. So automatic brain tumor image segmentation is required. The goal of segmentation is to outline the tumor, including its sub-compartments and surrounding tissues, whereas the main problem of registration and modeling is to deal with the tumor's morphological alterations. The advantages and disadvantages of various approaches are explored, with a focus on methods that may be used with routine clinical imaging protocols. Finally, a critical assessment of the existing condition is carried out, as well as prospective changes and trends, with a focus on recent developments in radiographic tumor assessment criteria. The proposed networks are suited to MRI images of glioblastomas (both low and high grade). We explore based on Convolutional Neural Networks (CNN), a tumor detection method on the basis of training and testing data set and Obtain tumor region of 90-95% accuracy.

TABLE OF CONTENTS

Page No.

CHAPTER 1: OVERVIEW OF BRAIN TUMOR SEGMENTATION9

1.1	Introduction	9
1.2	Literature Review.....	11
1.3	Brain Tumor Segmentation	11
1.3.1	MRI	12
1.3.2	MRI Work	12
1.3.3	MRI Images	13

CHAPTER 2: RELATED WORK14

CHAPTER 3: DEVELOPMENT TOOLS AND LIBRARIES17

3.1	Python Libraries	17
3.1.1	Numpy	17
3.1.2	Pandas	17
3.1.3	Matplotlib	17
3.1.4	TensorFlow	18
3.1.5	Keras	18
3.1.6	Scikit-Learn	18
3.1.7	SimpleITK	18
3.2	Development Tools	19
3.2.1	Jupyter Notebook	19

CHAPTER 4: FRAMEWORK METHODOLOGY20

4.1	Proposed Method.....	20
-----	----------------------	----

4.2	Pre-Processing	20
4.3	Feature Extraction.....	21
4.4	Convolution Neural Network.....	21
4.4.1	Initialization	22
4.4.2	Activation Function	22
4.4.3	Pooling	22
4.4.4	Regularization	23
4.4.5	Data Augmentation	23
4.4.6	Loss Function	23
4.5	Post-Processing.....	24
 CHAPTER 5: RESULT AND DISCUSSION		25
5.1	Dataset	25
5.2	Random MRI Image Visulaization.....	26
5.3	Segmented Area Prediction.....	27
5.4	Discussion Accuracy	33
 CHAPTER 6: CONCLUSION AND FUTURE WORK		37
6.1	Conclusion	37
6.2	Future work.....	37
 Reference		38

LIST OF FIGURES

Figure1.3: Brain Tumor Segmentation.....	12
Figure 1.3.3: MRI Images with tumor mask.....	13
Figure 4.1: Flow Diagram of Proposed Method.....	20
Figure 4.4: Architecture of Convolution Neural Network	22
Figure 5.2: Random MRI image visualization	26
Figure 5.2.1: Summary of Model	34
Figure 5.2.2: Training Result	35
Figure 5.2.3: Test Result	35
Figure 5.2.4: Accuracy graph of Training and Testing	36
Figure 5.2.5: Loss graph of Training and Testing.....	36

LIST OF TABLES

Table 5.3: Segmented Output Image.....	32
--	----

CHAPTER 1: OVERVIEW OF BRAIN TUMOR SEGMENTATION

1.1 Introduction

Primary malignant brain tumors are among the most dreadful types of cancer, not only because of the dismal prognosis, but also due to the direct consequences on decreased cognitive function and poor quality of life. The most frequent primary brain tumors in adults are primary central nervous system lymphomas and gliomas, which the latter account for almost 80% of malignant cases [1]. The term glioma encompasses many subtypes of the primary brain tumor, which range from slower-growing ‘low-grade’ tumors to heterogeneous, highly infiltrative malignant tumors. Despite significant advances in imaging, radiotherapy, chemotherapy and of malignant brain tumors, e.g., high-grade glioblastoma and metastasis, are still considered untreatable with a 2.5-year cumulative relative survival rate of 8% and 2% at 10 years [2]. Moreover, there are variable prognosis results for patients with low grade gliomas (LGG) with an overall 10-year survival rate about 57%. Previous studies have demonstrated that the magnetic resonance imaging (MRI) characteristics of newly identified brain tumors can be used to indicate the likely diagnosis and treatment strategy. Image segmentation is a critical step for the MRI images to be used in brain tumor studies: the segmented brain tumor extent can eliminate confounding structures from other brain tissues and therefore provide a more accurate classification for the subtypes of brain tumors and inform the subsequent diagnosis; the accurate delineation is crucial in radiotherapy or surgical planning, from which not only brain tumor extent has been outlined and surrounding healthy tissues has been excluded carefully in order to avoid injury to the sites of language, motor, and sensory function during the therapy; and segmentation of longitudinal MRI scans can efficiently monitor brain tumor recurrence, growth or shrinkage. In current clinical practice, the segmentation is still relied on manual delineation by human operators.

The manual segmentation is a very labor-intensive task, which normally involves slice-by-slice procedures, and the results are greatly dependent on operators’ experience and their subjective decision making. Moreover, reproducible results are difficult to achieve even by the same operator. For a multimodal, multi-institutional and longitudinal clinical trial, a fully automatic, objective and reproducible segmentation method is highly in demand. Despite recent developing in semi-automatic and fully automatic algorithms for brain tumor segmentation, there are still several opening challenges for this task mainly due to the high variation of brain tumors in size, shape, regularity, location and their heterogeneous appearance (e.g., contrast uptake, image uniformity and texture) [3].

However, HGG usually exhibit unclear and irregular boundaries that might also involve discontinuities due to aggressive tumor infiltration. This can cause problems and result in poor tumor segmentation; varies tumor sub-regions and tumor types can only be visible by considering multimodal MRI data. However, the co-registration across multiple MRI sequences can be difficult especially when these Automatic Brain Tumor Detection and Segmentation 507 sequences are acquired in different spatial resolutions; and typical clinical MRI images are normally acquired with higher in-plane resolution and much lower interslice resolution in order to balance between adequate image slices to cover the whole tumor volume with good quality cross-sectional views and the restricted scanning time. This can cause inadequate signal to noise ratio and asymmetrical partial volume effects may also affect the final segmentation accuracy. Previous studies on brain tumor segmentation can be roughly categorized into unsupervised learning based and supervised learning-based methods. A more detailed topical review on various brain tumor segmentation methods can be found elsewhere, e.g., in [4]. A study was proposed in which extremely randomized forest was used for classifying both appearance and context based features and 83% Dice score was achieved[4]. More recently, Soltaninejad et al. combined extremely randomized trees classification with super pixel based over-segmentation for a single FLAIR sequence based MRI scan that obtained 88% overall Dice score of the complete tumor segmentation for both LGG and HGG tumor cases. Nevertheless, the tuning of super pixel size and compactness could be tricky and influence the final delineation. Recently, supervised deep convolutional neural networks (CNN) have attracted lots of interests. Compared to conventional supervised machine learning methods, these deep learning based methods are not dependent on hand-crafted features, but automatically learn a hierarchy of increasingly complex features directly from data. Currently, using BRATS datasets and their benchmarking system, deep learning based methods have been ranked on top of the contest.

This can be attributed to the fact that deep CNN is constructed by stacking several convolutional layers, which involve convolving a signal or an image with kernels to form a hierarchy of features that are more robust and adaptive for the discriminative models. Despite recent advances in these deep learning based methods, there are still several challenges: 508 H. Dong et al. essentially tumor segmentation is an abnormal detection problem, it is more challenging than other pattern recognition based tasks; while most methods provided satisfied segmentation for HGG cases, in general the performance of the LGG segmentation is still poor; compared to complete tumor segmentation, the delineation of core tumor regions and enhanced infiltrative regions is still underperformed; a more computing-efficient and memory-efficient development is still in demand because existing CNN based methods require considerable amount of computing resources. In this study, we developed a novel 2D fully convoluted segmentation network that is based on the U-Net architecture[5].

1.2 Literature Review

Cancer can be detected as the uncontrolled, unnatural growth and division of the cells in the body. Occurrence, as a mass, of these unnatural cell growth and division in the brain tissue is called a brain tumor. Four standard MRI modalities used for glioma diagnosis include T1-weighted MRI (T1), T2-weighted MRI (T2), T1-weighted MRI with gadolinium contrast enhancement (T1-Gd) and Fluid Attenuated Inversion Recovery (FLAIR)[6]. Glioma is a tumor composed of a necrotic core, a margin of tumor activity, and edema tissue. These are grouped into two main categories based on their level of severity: low-grade glioma (LGG) and high-grade glioma (HGG)[7]. LGG is considered as benign whereas HGG is malignant. The image analysis deals with automatic or semi-automatic methods to help interpret the acquired images. Due to the large amount of data, which are currently being generated in the clinics, it is not possible to manually annotate and segment the data in a reasonable time. Image segmentation aims at partitioning an image into several segments. These segments can be chosen according to structures of interest, tissue types, functional areas, etc. Balafouti et al [8] presented a recent review targeted at brain segmentation specially. Image registration aims at aligning two different images in a common reference space. The segmentation is crucial for monitoring tumor growth or shrinkage in patients during therapy, for tumor volume measurements and it also plays an important role in surgical planning or radiotherapy planning, where not only the tumor has to be outlined, but also surrounding healthy structures are of interest.

1.3 Brain Tumor Segmentation

A brain tumor is a mass or growth of abnormal cells in your brain. Many different types of brain tumors exist. Some brain tumors are noncancerous (benign), and some brain tumors are cancerous (malignant). Brain tumors can begin in your brain (primary brain tumors), or cancer can begin in other parts of your body and spread to your brain (secondary, or metastatic, brain tumors). How quickly a brain tumor grows can vary greatly. The growth rate as well as location of a brain tumor determines how it will affect the function of your nervous system. Brain tumor treatment options depend on the type of brain tumor you have, as well as its size and location. Brain tumor segmentation is the process of separating the tumor from normal brain tissues; in clinical routine, it provides useful information for diagnosis and treatment planning. However, it is still a challenging task due to the irregular form and confusing boundaries of tumors. Tumor cells thermally represent a heat source; their temperature is high compared to normal brain cells. The main aim of the present paper is to demonstrate that thermal information of brain tumors can be used to reduce false positive and false negative results of segmentation performed in MRI images.

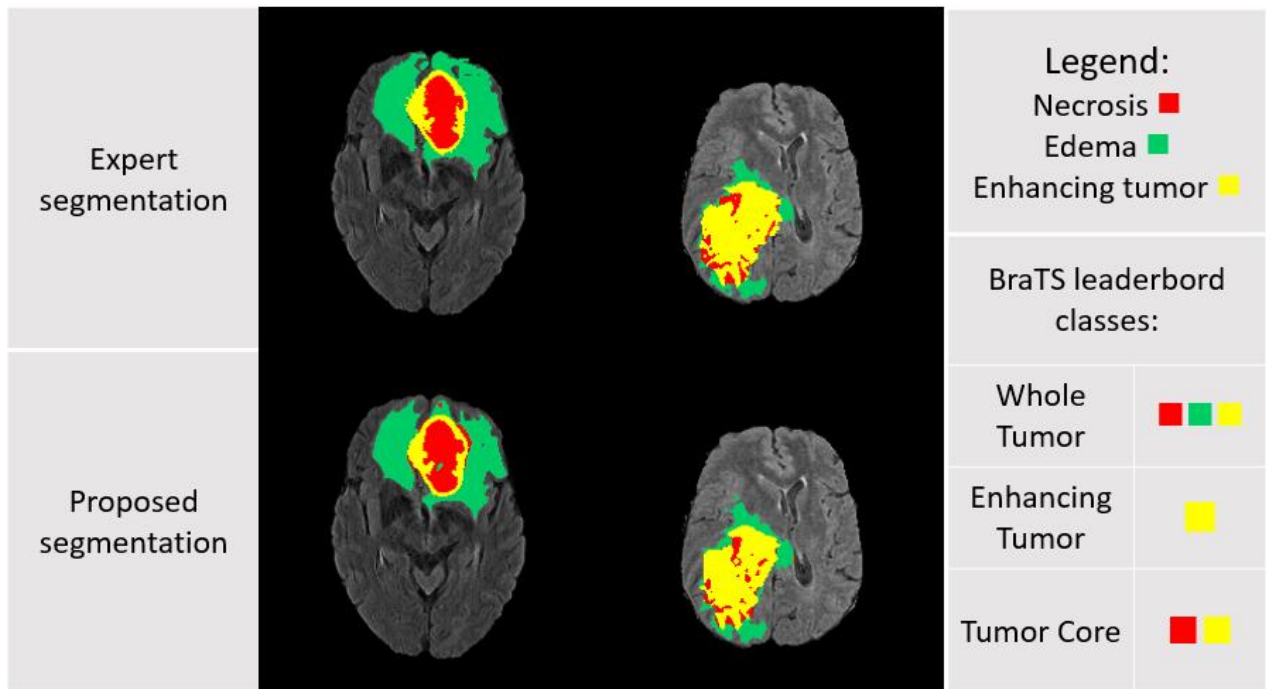


Figure1.3: Brain Tumor Segmentation[31]

Pennes bioheat equation was solved numerically using the finite difference method to simulate the temperature distribution in the brain; Gaussian noises of $\pm 2\%$ were added to the simulated temperatures.

1.3.1 MRI

Magnetic Resonance Imaging is a non-invasive imaging technology that produces three dimensional detailed anatomical images. It is often used for disease detection, diagnosis, and treatment monitoring. It is based on sophisticated technology that excites and detects the change in the direction of the rotational axis of protons found in the water that makes up living tissues.

1.3.2 MRI Work

MRIs employ powerful magnets which produce a strong magnetic field that forces protons in the body to align with that field. When a radiofrequency current is then pulsed through the patient, the protons are stimulated, and spin out of equilibrium, straining against the pull of the magnetic field. When the radiofrequency field is turned off, the MRI sensors are able to detect the energy released as the protons realign with the magnetic field. The time it takes for the protons to realign with the magnetic field, as well as the amount of energy released, changes depending on the environment and the chemical nature of the molecules. Physicians are able to tell the difference between various types of tissues based on these magnetic properties.

1.3.3 MRI Images

The most common MRI sequences are T1-weighted and T2-weighted scans. T1-weighted images are produced by using short TE and TR times. The contrast and brightness of the image are predominately determined by T1 properties of tissue. Conversely, T2-weighted images are produced by using longer TE and TR times. In these images, the contrast and brightness are predominately determined by the T2 properties of tissue. In general, T1- and T2-weighted images can be easily differentiated by looking at the CSF. CSF is dark on T1-weighted imaging and bright on T2-weighted imaging. A third commonly used sequence is the Fluid Attenuated Inversion Recovery (FLAIR). The FLAIR sequence is similar to a T2-weighted image except that the TE and TR times are very long. By doing so, abnormalities remain bright but normal CSF fluid is attenuated and made dark. This sequence is very sensitive to pathology and makes the differentiation between CSF and an abnormality much easier.

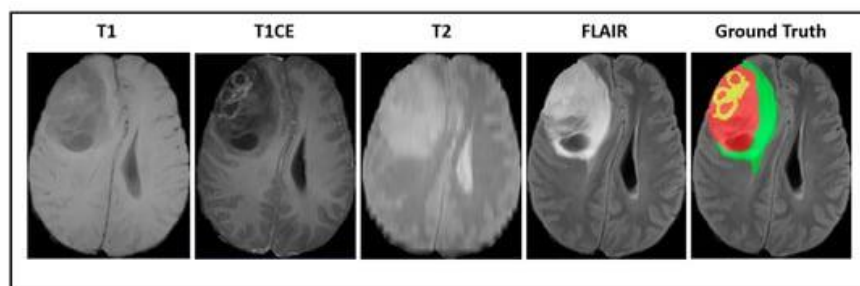


Figure 1.3.3: MRI Images with tumor mask[32]

CHAPTER 2: Related Work

As noted by Menze et al. [9], the number of publications devoted to automated brain tumor segmentation has grown exponentially in the last several decades. This observation not only underlines the need for automatic brain tumor segmentation tools, but also shows that research in that area is still a work in progress. Brain tumor segmentation methods (especially those devoted to MRI) can be roughly divided in two categories: those based on generative models and those based on discriminative models.

Recently performance of deep learning methods has attracted researchers to employ CNN for biological image segmentation. Additionally, for semantic segmentation, CNN and FCNN exhibited prominent segmentation accuracy. Most successful neural networks are based on encoder decoder based structures. Among these U-net and V-net based networks are popular architectures [10]. To obtain better results, important points to consider include preprocessing, data augmentation, batch normalization, loss function optimization function selection and postprocessing has shown the effectiveness of preprocessing and data augmentation in brain tumor segmentation process. Due to BRATS data format, few preprocessing operations are considered to be essential like pixel intensity range standardization [11] have used conditional random fields on predictions achieved by the network to fine-tune their results. Hierarchical network-based segmentation approaches like multi-resolution based loss calculation also exhibited acceptable result. Besides automatic approaches, interactive approaches also produce good results and being explored. There are three main aspects of neural networks which are being explored by researchers: First, extraction of useful features for classification or segmentation, second, finding relative importance of extracted features and third, developing right loss function for underlying problem. For the first task, different network architectures are being designed and tested like VGG, Inception, DeepLab, and Google-Net[12].

To give right importance to extracted features, different weighting schemes have been explored. This includes the weighing of features in loss layer computed at different granularity level like weighted cross entropy or weighted categorical cross entropy. The weighting schemes include the dataset based feature weighting/ class weighting using average, median, and maximum frequency based weights. Another approach is to calculate the features weights based on input batch similar to the proposed work[13]. Have used a gated Convolutional network where relative weights for each pixel are generated and applied in last Convolutional layer. Squeeze and Excitation block (employed in this work) provides another view to weight the computed feature maps in different phases of the network instead of applying them in the last Convolutional layer or loss layer. Defining or selecting right loss function is very important to boost up neural network performance. Normal loss functions like categorical cross entropy and least square methods provide good results for balanced to semi-balanced datasets however the datasets having large imbalance among the classes require special attention and specialized loss functions can perform better. It have provided a comparative study of different loss functions for highly unbalanced segmentation task and they argue that generalized dice similarity coefficient (DSC) function is better than others.[14] employed CNN with small (3x3) filters for deeper architecture to segment brain tumor in MRI and claimed

0.88, 0.93, 0.74 segmentation accuracy for whole tumor, core tumor and active tumor respectively on vBRATS dataset. Similarly, Havaei et al. [15] used Cascaded Two-pathway CNNs for simultaneous local and global processing of brain tumor identification and segmentation. Now a days the automatic segmentation uses deep learning techniques for segmentation. It provides efficient segmentation for a large amount of MRI based image data. The article reviewed the state of the art methods of deep learning. The conventional automatic segmentation methods need prior knowledge into probabilistic maps or selecting highly representative features for classifiers, which is challenging task. But, the convolutional neural network method automatically learns the corresponding complex features for both healthy brain tissues and tumour tissues from multimodal MRI brain images[16]. Cui et al., (2018) developed a novel automatic segmentation based on cascaded deep learning convolutional neural network. It has two sub networks; tumour localization network (TLN) and an intra tumour classification network (ITCN). The tumour region from the MRI brain slice is separated using tumour localization network and ITCN helps to label the defined tumour region into multiple sub-regions. The work was performed on multimodal brain tumour segmentation (BRATS, 2015) dataset, which had 220 high grade glioma (HGG) and 54 low grade glioma (LGG) cases. The evaluation can be performed by dice coefficient, positive predictive value (PPV) and sensitivity[17]. Typical classifier deep neural networks like AlexNet, VGGNet or GoogLeNet read in an image and output a set of class probabilities regarding the entire image. This is done by providing the raw image as input to the first layer of the network and performing a “forward pass”: Sequentially letting each layer compute its respective function on the output of the layer that preceded it. An example for a classification problem is handwritten digit recognition, where a neural network receives an image of a single handwritten digit and must decide which of the ten digits is depicted on the image.

Segmentation tasks, on the other hand, require a classification at every pixel of the input. This goal can be met by applying a classifier to patches of the input in a sliding window fashion. In this scheme, each pixel is classified by extracting a patch around it and having it classified by the network. Multiple pixels can be classified at once by processing multiple overlapping patches as a batch and parallelizing the forward pass of each patch through the network. As nearby pixels in an image are similar in their vicinity, this approach suffers from redundant computation. Another limitation of this method is that the learning process is restricted to features that are visible in each patch and the network has less of a chance to learn global features the smaller the patch size is. This is supported by reports of larger patch sizes yielding better results [18]. Different methods have been proposed to tackle the problem of creating a CNN that can generate a segmentation map for an entire input image in a single forward pass. Long et al [19]. Restore the down sampled feature maps to the original size of the input using deconvolution operations. Deconvolution is proposed by Zeiler and Fergus as a means of visualizing CNNs. The authors refer to it as backwards-strided convolution, using learnable kernels that are initialized to compute bilinear interpolation. The coarse segmentation map acquired through deconvolution is made finer by combining it with segmentation maps computed from earlier stages in the network, where the input has been down sampled fewer times. The authors refer to their network design as a “fully convolutional network” (FCN) [20].

Successful CNN-based medical image segmentation methods often draw on these recent findings in semantic segmentation. The two most common approaches are training a CNN on patches extracted from images and doing inference by sliding the CNN across all pixels of the network, predicting one pixel in each forward pass and training an FCN on full images or large sections of images. Certain aspects of medical images call for modifications on designs that are created for semantic segmentation, most notably the increased memory demands of 3D images and the imbalance between labels found in 3D ground truth data. [21]

CHAPTER 3: DEVELOPMENT TOOLS AND LIBRARIES

3.1 Python Libraries

Let's see some details about some python libraries that are useful for building our model:

3.1.1 Numpy

Numpy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level basic and advanced mathematical functions to operate on these arrays. Besides its obvious scientific uses, numpy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows numpy to seamlessly and speedily integrate with a wide variety of databases [22].

3.1.2 Pandas

Pandas is a popular Python-based data analysis toolkit. It is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language [22].

3.1.3 Matplotlib

Matplotlib is a Python package for data visualization. It allows easy creation of various plots, including line, scattered, bar, box, and radial plots, with high flexibility for refined styling and customized annotation. The versatile artist module allows developers to define basically any kind of visualization. For regular usage, Matplotlib offers a simplistic object oriented interface, the pyplot module, for easy plotting. Besides generating static graphics, Matplotlib also supports an interactive interface which not only aids in creating a wide variety of plots but is also very useful in creating web-based applications. Matplotlib is readily integrated into popular development environments, such as Jupyter Notebook, and it supports many more advanced data visualization packages.

3.1.4 TensorFlow

TensorFlow is an open-source library developed by Google primarily for deep learning applications. It also supports traditional machine learning. TensorFlow was originally developed for large numerical computations without keeping deep learning in mind. However, it proved to be very useful for deep learning development as well, and before Google open-sourced it. TensorFlow accepts data in the form of multi-dimensional arrays of higher dimensions called tensors. Multi-dimensional arrays are very handy in handling large amounts of data. TensorFlow works on the basis of data flow graphs that have nodes and edges. As the execution mechanism is in the form of graphs, it is much easier to execute TensorFlow code in a distributed manner across a cluster of computers while using GPUs. The next part of the What is TensorFlow tutorial focuses on why should we use TensorFlow.

3.1.5 Keras

Keras is the high-level API of TensorFlow 2: an approachable, highly-productive interface for solving machine learning problems, with a focus on modern deep learning. It provides essential abstractions and building blocks for developing and shipping machine learning solutions with high iteration velocity.

3.1.6 Scikit-Learn

Scikit-learn exposes a wide variety of machine learning algorithms, both supervised and unsupervised, using a consistent, task-oriented interface, thus enabling easy comparison of methods for a given application. Since it relies on the scientific Python ecosystem, it can easily be integrated into applications outside the traditional range of statistical data analysis. Importantly, the algorithms, implemented in a high-level language, can be used as building blocks for approaches specific to a use case.

3.1.7 SimpleITK

SimpleITK is a simplified programming interface to the algorithms and data structures of the Insight Toolkit (ITK) for segmentation, registration and advanced image analysis. Combining SimpleITK's Python bindings with the Jupyter notebook web application creates an environment which facilitates collaborative development of biomedical image analysis workflows.

3.2 Development Tools

3.2.1 Jupyter Notebook

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

CHAPTER 4: FRAMEWORK METHODOLOGY

4.1 Proposed Method

In our proposed method from MRI dataset we preprocessed dataset and split it by training and testing images. We trained our training images with CNN and any loss goes to loss function then CNN. Training CNN result or weight goes to testing CNN and compare with it. Finally predicted segmented tumor region.

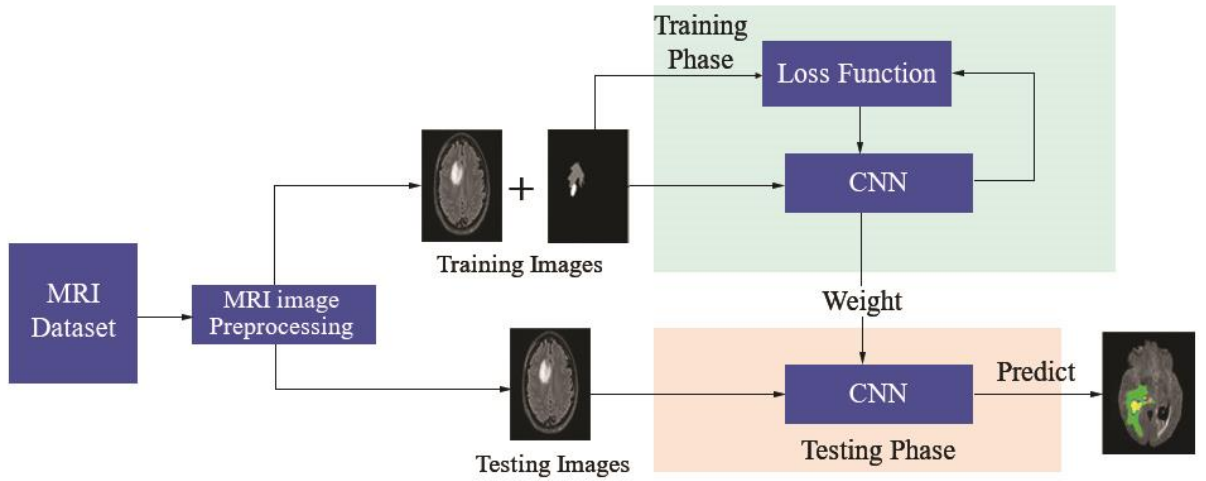


Figure 4.1: Flow diagram of proposed Method

4.2 Pre-Processing

Most algorithms rely on some kind of pre-processing for image preparation and image enhancement. Image denoising is a standard pre-processing task for MRI. MRI images are altered by the bias distortion. This makes the intensity of the same tissues to vary across the image. To correct it, we applied the N4ITK method[6]. The main challenging task is removing artifacts produced by inhomogeneity in a magnetic or small movements created by the patient during scanning. Many times bias is present in the scanning results, which affects the segmentation results, particularly in the computer-aided models. Chmelika et al., (2018) demonstrated work uses n4ITK bias correction for the T1 and T1C images in the data set. The n4ITK bias correction removes the intensity gradient on each scanning image. Noise reduction is also performed by median filter in order to standardize the pixel intensities. Hence, noise reduction and bias correction helps to improve the data processing and provides better segmentation. Multiple radio frequency pulse sequences can be used to provide the different types of tissue[23].

4.3 Feature Extraction

The modality used for feature extraction is dependent on the intrinsic properties of the tumor subregion. For example, edema features are extracted from FLAIR modality, since it is typically depicted by hyper-intense signal in FLAIR. Non-enhancing solid core features are extracted from T1Gd modality, since the appearance of the necrotic (NCR) and the non-enhancing (NET) tumor core is typically hypo-intense in T1Gd when compared to T1. Necrotic/cystic core tumor features are extracted from T1Gd modality, since it is described by areas that show hyper-intensity in T1Gd when compared to T1. The features we extracted can be grouped into three categories. The first category is the first order statistics, which includes maximum intensity, minimum intensity, mean, median, 10th percentile, 90th percentile, standard deviation, variance of intensity value, energy, entropy, and others. These features characterize the gray level intensity of the tumor region[24].

4.4 Convolutional Neural Networks

CNN were used to achieve some breakthrough results and win well-known contests. The application of convolutional layers consists in convolving a signal or an image with kernels to obtain feature maps. So, a unit in a feature map is connected to the previous layer through the weights other kernels. The weights of the kernels are adapted during the training phase by backpropagation, in order to enhance certain characteristics of the input. Since the kernels are shared among all units of the same feature maps, convolutional layers have fewer weights to train than dense FC layers, making CNN easier to train and less prone to overfitting. Moreover, since the same kernel is convolved over all the image, the same feature is detected independently of the location| translation invariance. By using kernels, information of the neighborhood is taken into account, which is an

useful source of context information. Usually, a non-linear activation function is applied on the output of each neural unit. If we stack several convolutional layers, the extracted features become more abstract with the increasing depth. The first layers enhance features such as edges, which are aggregated in the following layers as motifs, parts, or objects[25]. The following concepts are important in the context of CNN:

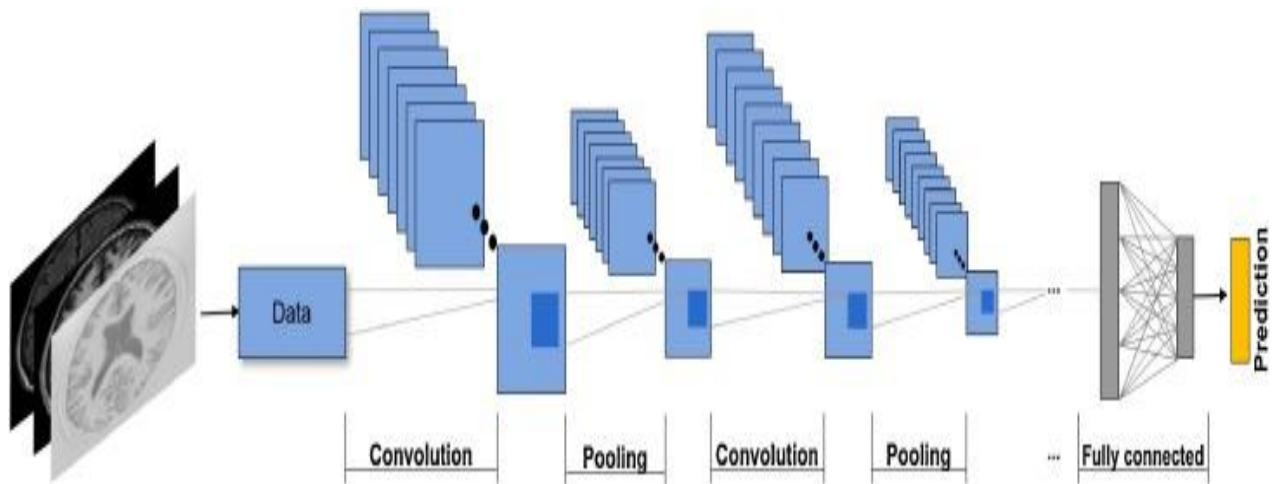


Figure 4.4: Architecture of Convolution Neural Network

4.4.1 Initialization

It is important to achieve convergence. We use the Xavier initialization. With this, the activations and the gradients are maintained in controlled levels, otherwise back-propagated gradients could vanish or explode.

4.4.2 Activation Function

It is responsible for non-linearly transforming the data. Rectifier linear units (ReLU), defined as were found to achieve better results than the more classical sigmoid, or hyperbolic tangent functions, and speed up training. However, imposing a constant 0 can impair the gradient flowing and consequent adjustment of the weights. We cope with these limitations using a variant called leaky rectifier linear unit (LReLU) that introduces a small slope on the negative part of the function. This function is defined as where is the leakyness parameter. In the last FC layer, we use softmax [26].

4.4.3 Pooling

Pooling It combines spatially nearby features in the feature maps. This combination of possibly redundant features makes the representation more compact and invariant to small image changes, such as insignificant details; it also decreases the computational load of the next stages. To join features it is more common to use max-pooling or average-pooling [27]

4.4.4 Regularization

It is used to reduce overfitting. We use Dropout in the FC layers. In each training step, it removes nodes from the network with probability . In this way, it forces all nodes of the FC layers to learn better representations of the data, preventing nodes from co-adapting to each other. At test time, all nodes are used. Dropout can be seen as an ensemble of different networks and a form of bagging, since each network is trained with a portion of the training data[28].

4.4.5 Data Augmentation

It can be used to increase the size of training sets and reduce overfitting. The purpose of data augmentation is to improve the network performance by intentionally producing more training data from the original one. In this study, we applied a set of data augmentation methods summarized in Table 1. Simple transformation such as flipping, rotation, shift and zoom can result in displacement fields to images but will not create training samples with very different shapes. Shear operation can slightly distort the global shape of tumor in the horizontal direction, but is still not powerful to gain sufficient variable training data, as tumors have no definite shape. To cope with this problem, we further applied elastic distortion [29] that can generate more training data with arbitrary but reasonable shapes.

Table 1. Summary of the applied data augmentation methods (γ controls the brightness of the outputs; α and σ control the degree of the elastic distortion).

Methods	Range
Flip horizontally	50% probability
Flip vertically	50% probability
Rotation	$\pm 20^\circ$
Shift	10% on both horizontal and vertical direction
Shear	20% on horizontal direction
Zoom	$\pm 10\%$
Brightness	$\gamma = 0.8-1.2$
Elastic distortion	$\alpha = 720, \sigma = 24,$

4.4.6 Loss Function

It is the function to be minimized during training. We used the Categorical Cross-entropy, where represents the probabilistic predictions (after the softmax) and is the target. In the next subsections, we discuss the architecture and training of our CNN

4.5 Post-Processing

The post-processing step is performed to further renew the segmentation results. However, these techniques are computationally expensive. Connected components analysis involves finding and extracting connected components and then applying a simple thresholding technique to remove unwanted blobs. Another technique of removing false positive around edges of the segmentation image is to apply morphological operations, erosion, and dilation in succession.

Ch-5: Result and Discussion

5.1 Dataset:

Dataset used in: LGG Segmentation Dataset

Mateusz Buda, Ashirbani Saha, Maciej A. Mazurowski "Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm." Computers in Biology and Medicine, 2019. Maciej A. Mazurowski, Kal Clark, Nicholas M. Czarnek, Parisa Shamsesfandabadi, Katherine B. Peters, Ashirbani Saha "Radiogenomics of lower-grade glioma: algorithmically-assessed tumor shape is associated with tumor genomic subtypes and patient outcomes in a multi-institutional study with The Cancer Genome Atlas data." Journal of Neuro-Oncology, 2017. This dataset contains brain MRI images together with manual FLAIR abnormality segmentation masks. The images were obtained from The Cancer Imaging Archive (TCIA). They correspond to 110 patients included in The Cancer Genome Atlas (TCGA) lower-grade glioma collection with at least fluid-attenuated inversion recovery (FLAIR) sequence and genomic cluster data available. Tumor genomic clusters and patient data is provided in data.csv file.

5.2 Random MRI images visualization

For the brain tumor segmentation challenge, we proposed a model and CNN-based fully convolutional networks. Tumor detection and segmentation are essentially part of the semantic segmentation problem. In our dataset all the images are resize by 256×256 means height and weight respectively 256 with channel size 3. **Masking** is an image processing method in which we define a small 'image piece' and use it to modify a larger image. **Masking** is the process that is underneath many types of image processing, including edge detection, motion detection, and noise reduction.

So we loaded the images path and mask path and appended the mask files to the trained files. Figure below shown the Random MRI images visualization

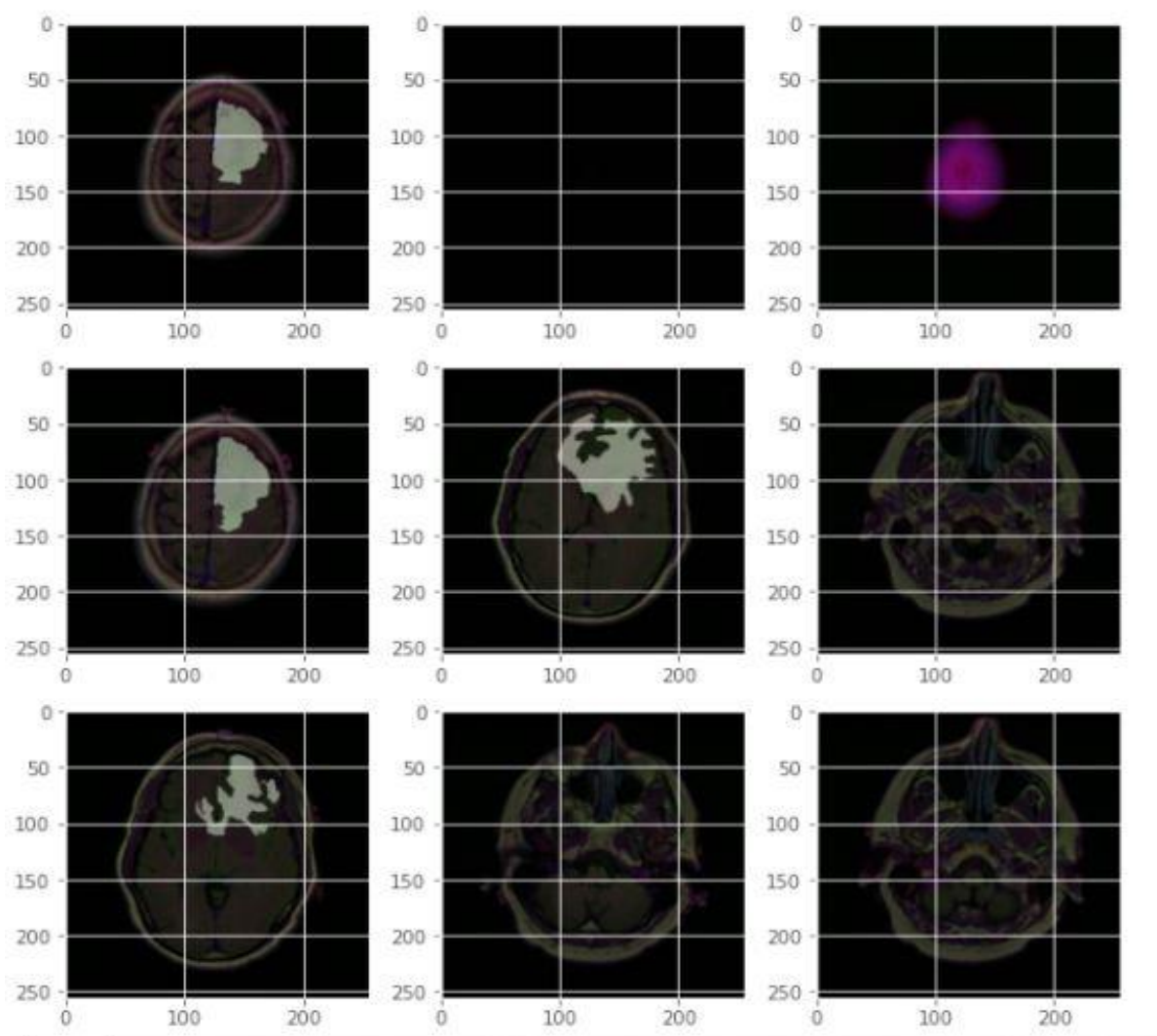
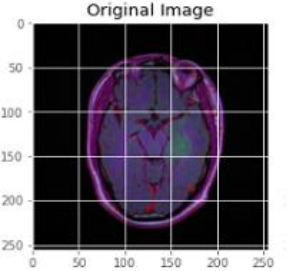
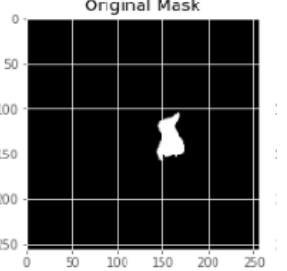
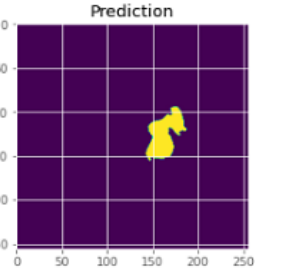
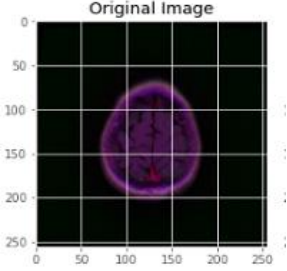

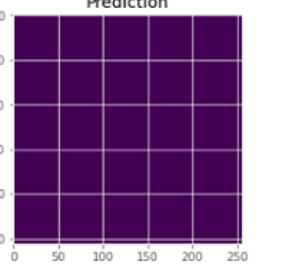
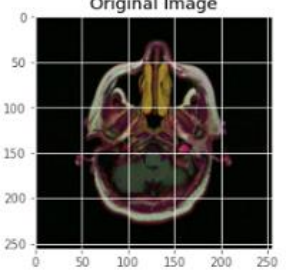

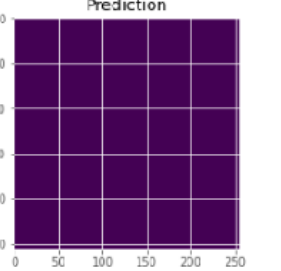
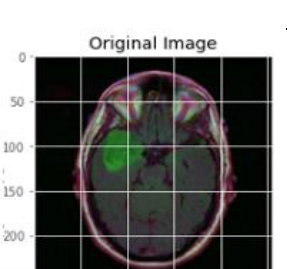
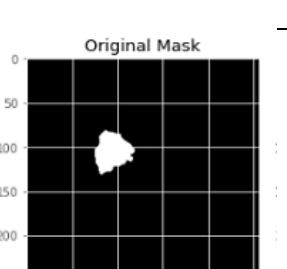
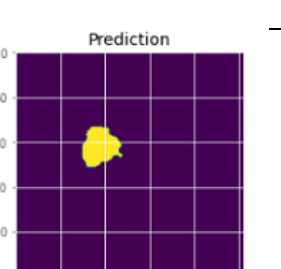
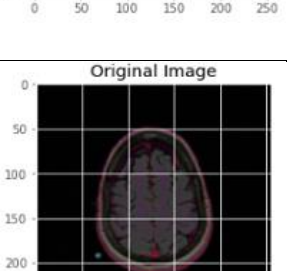
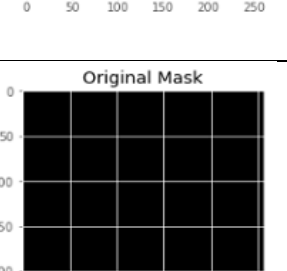
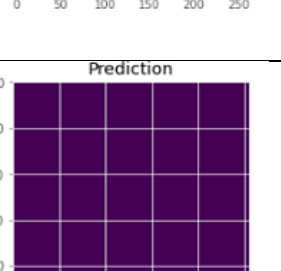
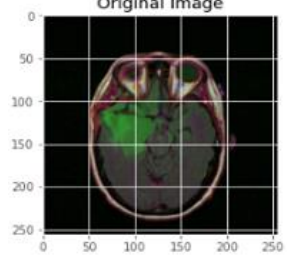
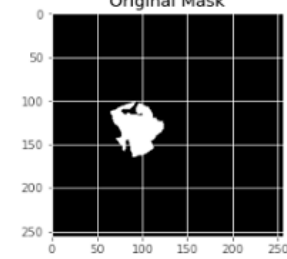
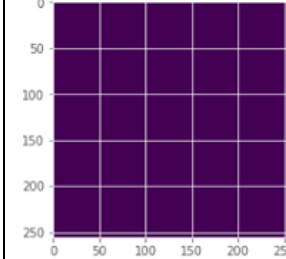
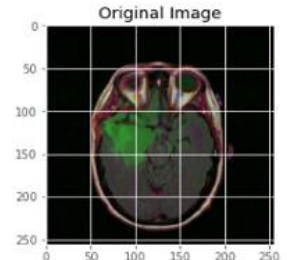
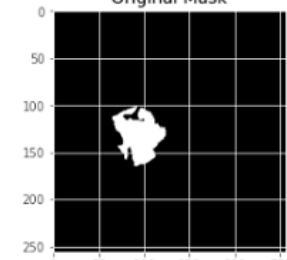
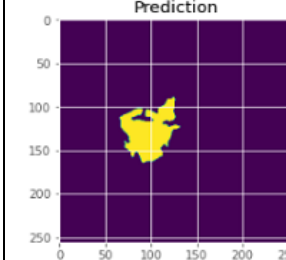
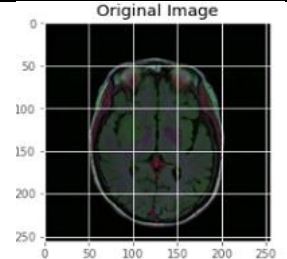
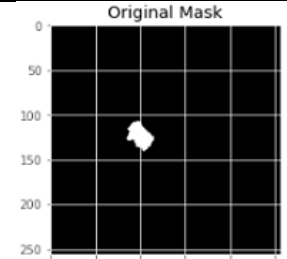
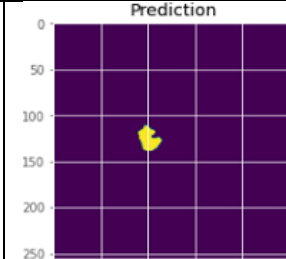
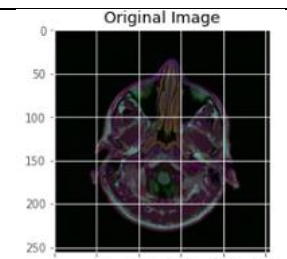
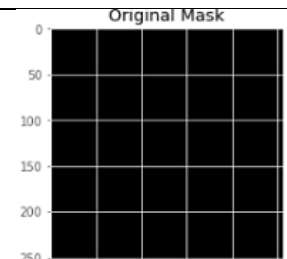
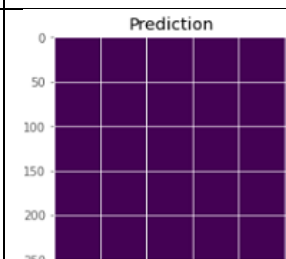
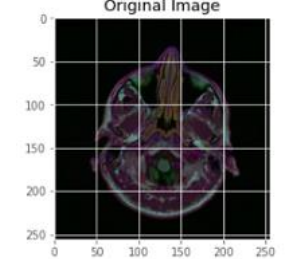
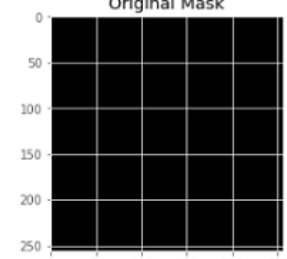
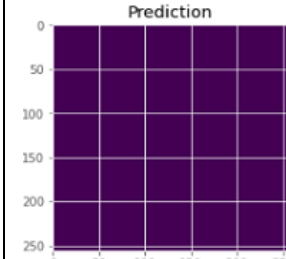
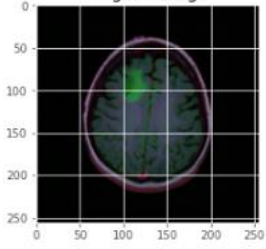
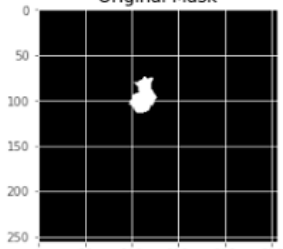
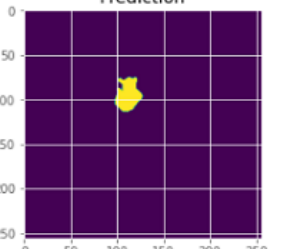
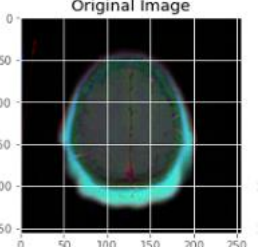
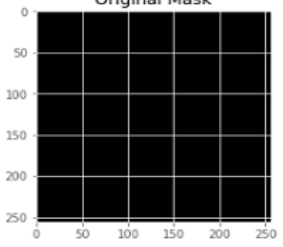
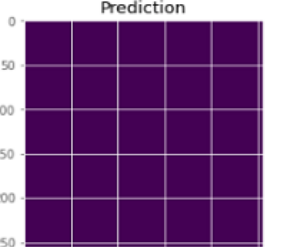
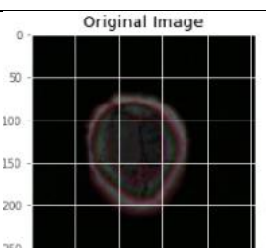
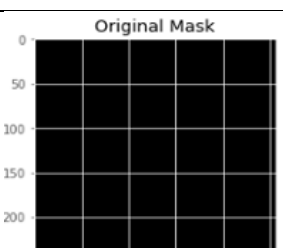
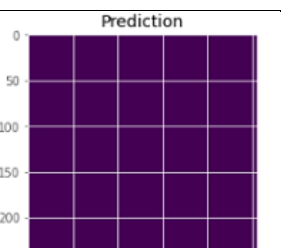
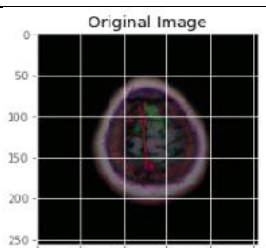
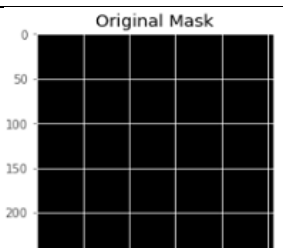
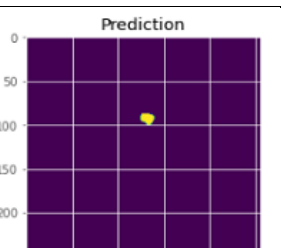
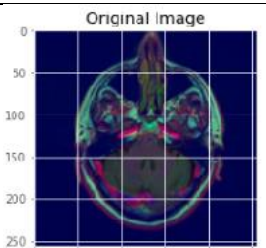
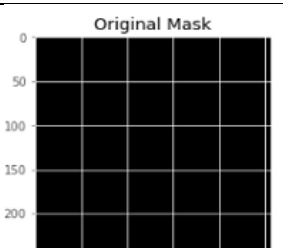
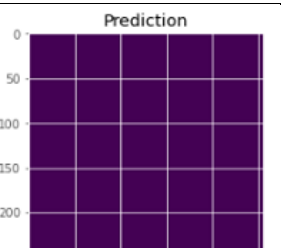


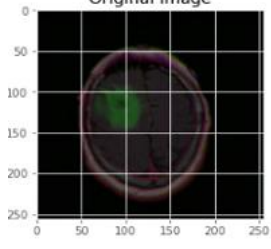
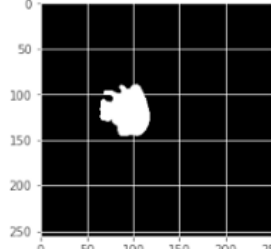
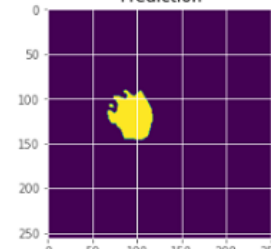
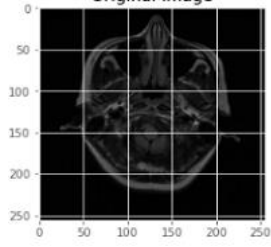
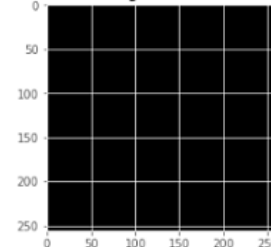
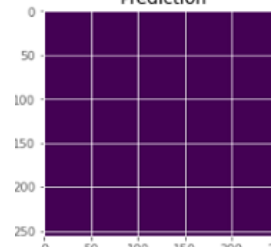
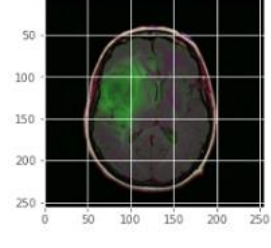
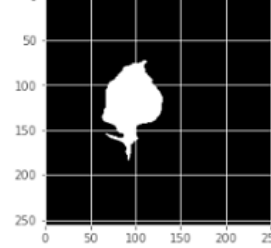
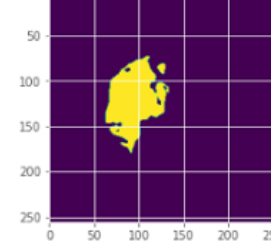
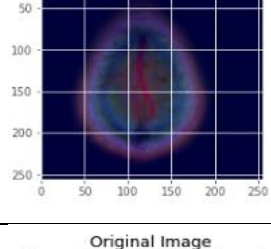
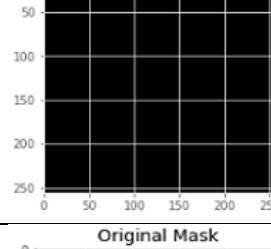
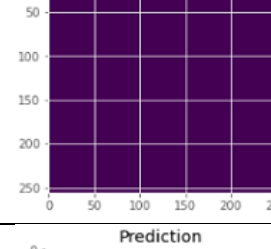
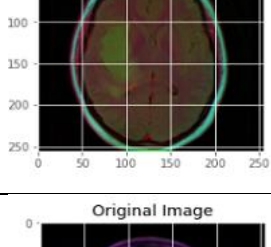
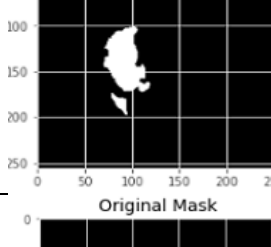
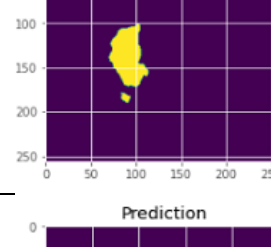
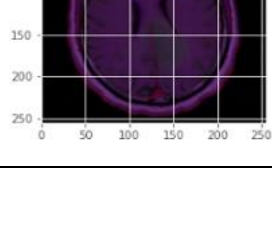
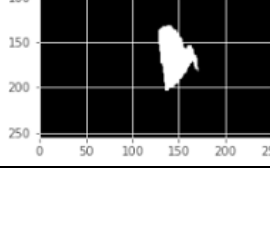
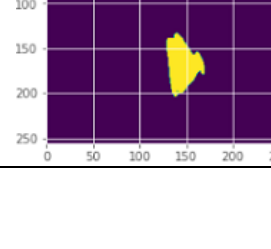
Figure 5.2: Random MRI images visualization

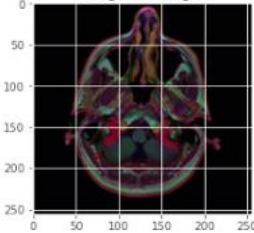
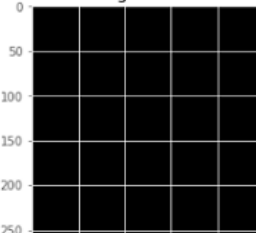
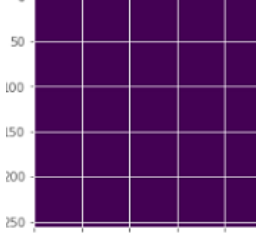
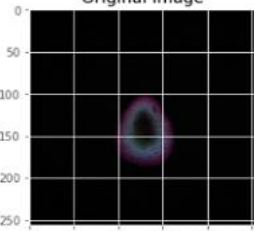
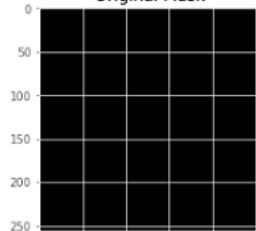
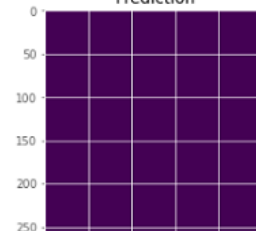
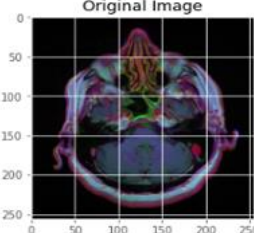
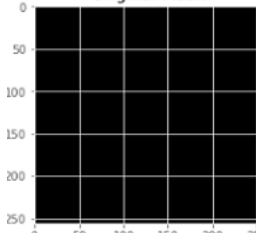
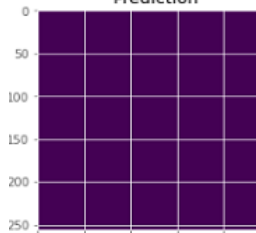
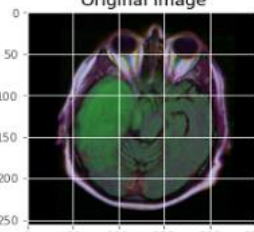
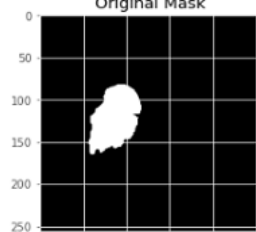
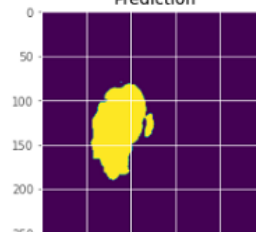
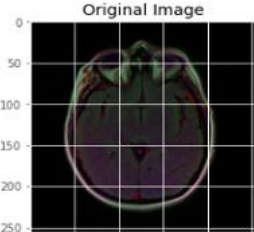
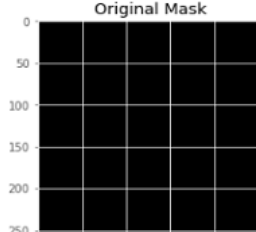
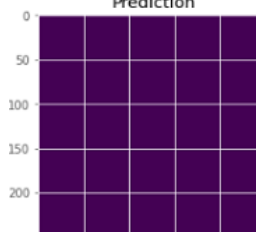
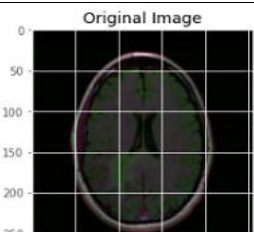
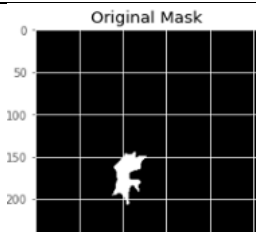
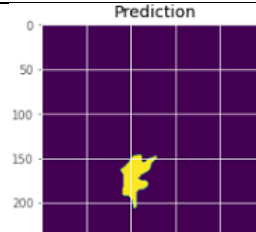
5.3 Segmented Area Prediction

Images No	Original Image	Original Mask	Prediction
01			
02			
03			
04			
05			

06	<p>Original Image</p> 	<p>Original Mask</p> 	<p>Prediction</p> 
07	<p>Original Image</p> 	<p>Original Mask</p> 	<p>Prediction</p> 
08	<p>Original Image</p> 	<p>Original Mask</p> 	<p>Prediction</p> 
09	<p>Original Image</p> 	<p>Original Mask</p> 	<p>Prediction</p> 
10	<p>Original Image</p> 	<p>Original Mask</p> 	<p>Prediction</p> 

11	<p>Original Image</p> 	<p>Original Mask</p> 	<p>Prediction</p> 
12	<p>Original Image</p> 	<p>Original Mask</p> 	<p>Prediction</p> 
13	<p>Original Image</p> 	<p>Original Mask</p> 	<p>Prediction</p> 
14	<p>Original Image</p> 	<p>Original Mask</p> 	<p>Prediction</p> 
15	<p>Original Image</p> 	<p>Original Mask</p> 	<p>Prediction</p> 

16			
17			
18			
19			
20			
21			

22			
23			
24			
25			
26			
27			

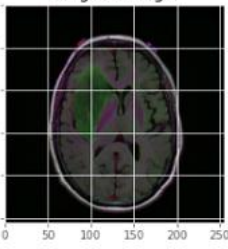
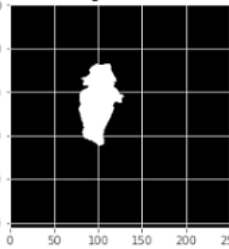
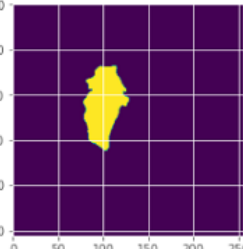
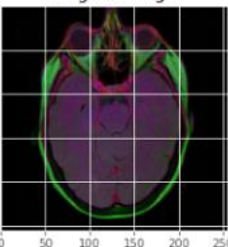
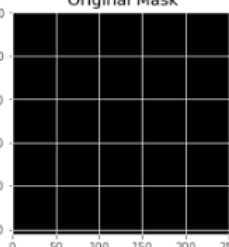
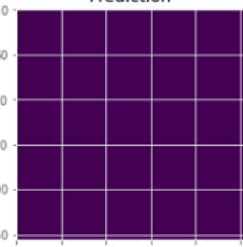
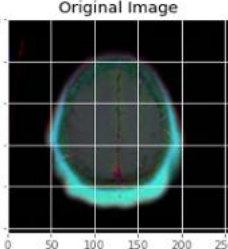
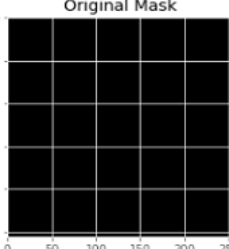
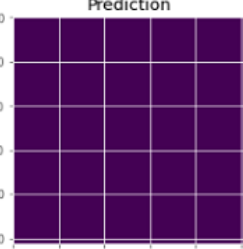
28			
29			
30			

Table 5.3: Segmented Output Image

Using MRI image segmentation process we achieve overall result 30 images from randomly process. We got 13 images have been detected tumor and others images non-tumor detection because of their original images are not tumor. Finally we see the predicted from original image here

Eg. see with binary value for prediction:

1=Yes

0=NO

Original image	Original mask	prediction	Decision
1	1	we get tumor (Yes)	
0	0	we didn't get tumor (No)	

Input image :30
Tumor predict:13

5.4 Discussion of Accuracy:

We generate image and mask at the same time use the same seed for image_datagen and mask_datagen to ensure the transformation for image and mask is the same if you want to visualize the results of generator. We adjust the image and mask divided by 255 and assigned value either 0 or 1.

Rather of using a cross-entropy-based or quadratic cost function, the Soft Dice metric was described and used as the network's cost function during the training process. The Soft Dice Similarity Coefficient can be thought of as a differentiable version of the original Dice Similarity Coefficient (DSC). To optimize the cost function with regard to its parameters, deep neural networks use stochastic gradient-based optimization. To estimate the parameters, we used the adaptive moment estimator (Adam). In general, Adam uses the first and second moments of gradients to update and fix the current gradients' moving average. The learning rate was set to 0.0001 and the maximum number of epochs was set to 150 in our Adam optimizer. All biases were set to zero, and all weights were set to a normal distribution with a mean of 0 and a standard deviation of 0.01.

The DSC provides the overlap measurement between the manual delineated brain tumoral regions and the segmentation results of our fully automatic method that is

$$DSC = \frac{2TP}{FP + 2TP + FN}, \quad (1)$$

in which TP, FP and FN denote the true positive, false positive and false negative measurements, respectively [30]

The Intersection over Union (**IoU**) metric, also referred to as the Jaccard index, is essentially a method to quantify the percent overlap between the target mask and our prediction output. the IoU metric measures the number of pixels common between the target and prediction masks divided by the total number of pixels present across *both* masks.

$$IoU = \frac{target \cap prediction}{target \cup prediction}$$

In our CNN we used keras Conv2D for training and testing. We give the input 256 by 256 with channel size 3 and filter size 3 by 3 with same padding size and applied batch normalization technique for training very deep neural Networks that standardizes the input to a layer for each mini-batch. This has the effect of stabilizing the learning process and drastically reducing the number of training epochs required to deep networks and we used Relu activation function and maxpooling. Then we get Total params 31,043,521 Trainable params:31,037,633 and Non-trainable params:5,888

```
Model: "model"
```

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 256, 256, 3) 0		
conv2d (Conv2D)	(None, 256, 256, 64) 1792		input_1[0][0]
activation (Activation)	(None, 256, 256, 64) 0		conv2d[0][0]
conv2d_1 (Conv2D)	(None, 256, 256, 64) 36928		activation[0][0]
batch_normalization (BatchNorma	(None, 256, 256, 64) 256		conv2d_1[0][0]
activation_1 (Activation)	(None, 256, 256, 64) 0		batch_normaliza tion[0][0]

Figure 5.2.1: Summary of Model

For training and test we used epochs =150 Batch size=32 and learning rate 0.0001 decay rate is learning rate divided by epochs. We get dice coefficient value and loss, binary accuracy, IOU respectively 0.9067, 0.9984 and 0.8314 for training .

```

89/88 [=====] - 87s 977ms/step - loss: -0.9189 - binary
_accuracy: 0.9983 - iou: 0.8543 - dice_coef: 0.9184 - val_loss: -0.8956 - val_bi
nary_accuracy: 0.9984 - val_iou: 0.8132 - val_dice_coef: 0.8954
Epoch 148/150
89/88 [=====] - ETA: 0s - loss: -0.9227 - binary_accu
cy: 0.9984 - iou: 0.8591 - dice_coef: 0.9228
Epoch 00148: val_loss did not improve from -0.90675
89/88 [=====] - 87s 981ms/step - loss: -0.9227 - binary
_accuracy: 0.9984 - iou: 0.8591 - dice_coef: 0.9228 - val_loss: -0.9038 - val_bi
nary_accuracy: 0.9983 - val_iou: 0.8325 - val_dice_coef: 0.9073
Epoch 149/150
89/88 [=====] - ETA: 0s - loss: -0.9228 - binary_accu
cy: 0.9984 - iou: 0.8599 - dice_coef: 0.9230
Epoch 00149: val_loss did not improve from -0.90675
89/88 [=====] - 86s 971ms/step - loss: -0.9228 - binary
_accuracy: 0.9984 - iou: 0.8599 - dice_coef: 0.9230 - val_loss: -0.8914 - val_bi
nary_accuracy: 0.9983 - val_iou: 0.8048 - val_dice_coef: 0.8897
Epoch 150/150
89/88 [=====] - ETA: 0s - loss: -0.9209 - binary_accu
cy: 0.9984 - iou: 0.8526 - dice_coef: 0.9169
Epoch 00150: val_loss did not improve from -0.90675
89/88 [=====] - 88s 985ms/step - loss: -0.9209 - binary
_accuracy: 0.9984 - iou: 0.8526 - dice_coef: 0.9169 - val_loss: -0.9030 - val_bi
nary_accuracy: 0.9984 - val_iou: 0.8314 - val_dice_coef: 0.9067

```

Figure 5.2.2: Training Result

When we tested with 393 validated images then we get

test loss : -0.017

Test IOU: 0.9981

Test dice coefficient: 0.8477

```

Found 393 validated image filenames.
Found 393 validated image filenames.
13/12 [=====] - 4s 288ms/step - los
s: -0.9180 - binary_accuracy: 0.9981 - iou: 0.8477 - dice_coe
f: 0.9172
Test lost: -0.917961597442627
Test IOU: 0.9981355667114258
Test Dice Coefficient: 0.8477444052696228

```

Figure 5.2.3: Test Result

In our graph we saw the loss graph and accuracy graph where test loss for blue color and train loss for Red color as well as For loss graph train dice for red color and test dice for blue. We pick random image for range 30 from data set to plot the image segmentation. Original mask and predicted mask.

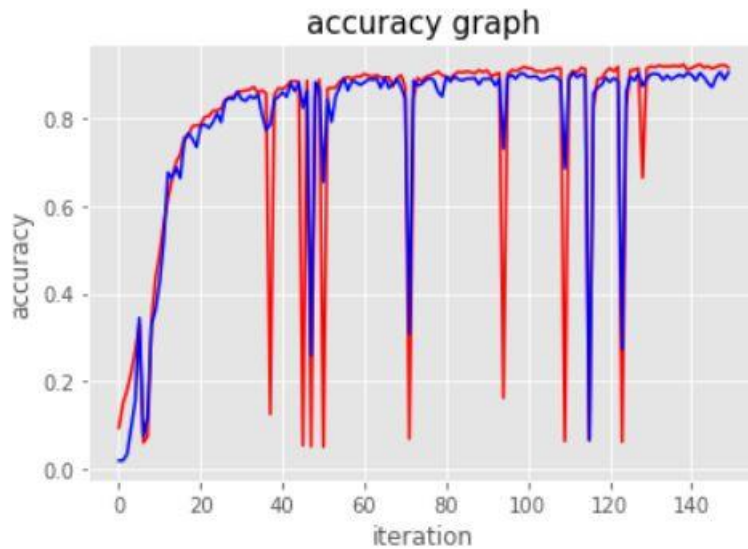


Figure 5.2.4: Accuracy Graph of Training & Testing

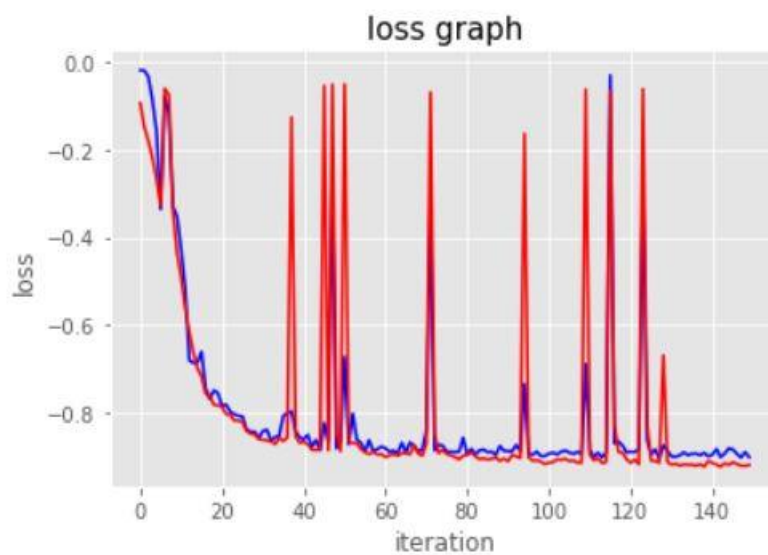


Figure 5.2.5: Loss Graph of Training & Testing

CHAPTER 6: CONCLUSION AND FUTURE WORK

6.1 Conclusion

We have described our research on brain tumor segmentation MRI images using Convolutional neural networks. In this paper, a fully automatic and accurate method for segmentation of whole brain tumor regions using a well known architecture in medical imaging called CNN used. Because it help in proper segmentation, the algorithm connects both local and global features. Using max pooling and drop out to complement the learning process, the training and testing speed is boosted. The fully connected layer's speed is improved by lowering the features. Overfitting is also reduced when parameters are reduced. We have obtained dice coefficient value 0.84. The results suggest that the applied strategy help in the detection of enhancing tumors as well as limiting tumor definition to the real tumor location.

6.2 Future Work

A more powerful GPU is planned for future work in order to speed up the learning phase of CNN. As a result, a greater number of CNN topologies, as well as other data augmentation approaches, can be evaluated. Another intriguing viewpoint is the usage of ensembles. Stacking and Blending are two learning approaches for improving segmentation performance in two types of tumors: active tumors and tumor cores and refine the segmentation results by lowering the false-positive rate using postprocessing. Also calculated the segmentation mask and predicted mask of resultant images can be worth.

References

- [1] Schwartzbaum, J.A., Fisher, J.L., Aldape, K.D., Wrensch, M.: Epidemiology and molecular pathology of glioma. *Nat. Clin. Pract. Neurol.* 2, 494–503 (2006)
- [2] Smoll, N.R., Schaller, K., Gautschi, O.P.: Long-term survival of patients with glioblastoma multiforme (GBM). *J. Clin. Neurosci.* 20, 670–675 (2013).
- [3] Hao Dong, Guang Yang, Fangde Liu, Yuanhan Mo, and Yike Guo. Auto-matic brain tumor detection and segmentation using u-net based fully con-volutional networks. In annual conference on medical image understanding and analysis, pages 506–517. Springer, 2017.
- [4] Pinto, A., Pereira, S., Correia, H., Oliveira, J., Rasteiro, D.M.L.D., Silva, C.A.: Brain tumour segmentation based on extremely randomized forest with high-level features. In: 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society(EMBC), pp. 3037–3040 (2015)
- [5] Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). doi:10.1007/978-3-319-24574-4_28
- [6] Ali I, sin, Cem Direko ğlu, and Melike S ah. Review of mri-based brain tumor image segmentation using deep learning methods. *Procedia Computer Science*, 102:317–324, 2016.
- [7] K Chethan and R Bhandarkar. An efficient medical image retrieval and classification using deep neural network. *Indian Journal of Science and Technology*, 13(39):4127–4141, 2020.
- [8] Mohd Ali Balafar, Abdul Rahman Ramli, M Iqbal Saripan, and Syamsiah Mashohor. Review of brainmri image segmentation methods. *Artificial Intelligence Review*, 33(3):261–274, 2010.
- [9] Bjoern H Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, et al. The multimodal brain tumor image segmentation benchmark (brats). *IEEE transactions on medical imaging*, 34(10):1993–2024, 2014.
- [10] M Havaei, A Davy, D Warde-Farley, A Biard, A Courville, Y Bengio, C Pal, PM Jodoin, and H Larochelle. Brain tumor segmentation with deep neural networks. *medical image analysis*. 2017.
- [11] Zhao, X., Wu, Y., Song, G., Li, Z., Zhang, Y., & Fan, Y. (2018). A deep learning model integrating FCNNs and CRFs for brain tumor segmentation. *Medical Image Analysis*, 43, 98–111.
- [12] Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A. (2017). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In *AAAI* (pp. 4278–4284).

- [13] Wang, G., Li, W., Zuluaga, M. A., Pratt, R., Patel, P. A., Aertsen, M. . . Vercauteren, T. (2017). Interactive medical image segmentation using deep learning with image-specific fine-tuning. arXiv preprint arXiv: 1710.04043.
- [14] Pereira, S., Pinto, A., Alves, V., & Silva, C. A. (2016). Brain tumor segmentation using convolutional neural networks in MRI images. *IEEE Transactions on Medical Imaging*, 35(5), 1240–1251.
- [15] Dong, H., Yang, G., Liu, F., Mo, Y., & Guo, Y. (2017). Automatic Brain Tumor Detection and Segmentation Using U-Net Based Fully Convolutional Networks. *Medical Image Understanding and Analysis (MIUA)2017*.
- [16] Ism A, Direkoglu C, Sah M (2016). In proceedings of 12th International Conference on Application of Fuzzy Systems and Soft Computing, August 29-30, Review of MRI based brain tumor image segmentation using deep learning methods, Vienna, Austria.
- [17] Cui S, Mao L, Jiang J, Liu C, Xiong S (2018). Automatic semantic segmentation of brain gliomas from MRI images using a deep cascaded neural network. *J Healthcare Engineering*, 2018, 1-14.
- [18] D. C. Cireş,an, A. Giusti, L. M. Gambardella, and J. Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. In *NIPS*, 2012.
- [19] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. 2015, arXiv:1411.4038v2 [cs.CV].
- [20] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. 2013, arXiv:1311.2901v3 [cs.CV].
- [21] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. 2015, arXiv:1505.04597v1 [cs.CV].
- [22] Wes McKinney. *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*. " O'Reilly Media, Inc.", 2012.
- [23] M Malathi and P Sinthia. Brain tumor segmentation using convolutional neural network with tensorflow. *Asian Pacific journal of cancer prevention: APJCP*, 20(7):2095, 2019.
- [24] M Mohammed Thaha, K Pradeep Mohan Kumar, BS Murugan, S Dhanasek-eran, P Vijaya karthick, and A Senthil Selvi. Brain tumor segmentation using convolutional neural networks in mri images. *Journal of medical systems*, 43(9):1–10, 2019.

- [25] S'ergio Pereira, Adriano Pinto, Victor Alves, and Carlos A Silva. Brain tumor segmentation using convolutional neural networks in mri images. *IEEE transactions on medical imaging*, 35(5):1240–1251, 2016.
- [26] K. Jarrett et al., “What is the best multi-stage architecture for object recognition?,” in *Proc. 12th Int. Conf. IEEE Comput. Vis.*, 2009, pp. 2146–2153.
- [27] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [28] G. E. Hinton et al., Improving neural networks by preventing co-adaptation of feature detectors *ArXiv Preprint arXiv:1207.0580v1*, 2012 [Online].
- [29] Milletari, F., Navab, N., Ahmadi, S.-A.: V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. *arXiv*, pp. 1–11 (2016)
- [30] L. R. Dice, “Measures of the amount of ecologic association between species,” *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.
- [31] <https://software.intel.com/content/dam/develop/external/us/en/images/-p-779095.png>
- [32] https://www.mdpi.com/applsci/applsci-10-07790/article_deploy/html/images/applsci-10-07790-g002-550.jpg