GIRIJANANDA CHOWDHURY INSTITUTE OF MANAGEMENT AND TECHNOLOGY, GUWAHATI

PROJECT REPORT

ON

"SEMANTIC SEGMENTATION OF BRAIN TUMOR ON MULTI-BAND 2D STACKED-UP SLICES USING ATTENTION-BASED U-NET"

Submitted in fulfillment of the requirement of the degree of
Bachelor of Technology
in
COMPUTER SCIENCE AND ENGINEERING
(2019-23)

ASSAM SCIENCE AND TECHNOLOGY UNIVERSITY, GUWAHATI

Submitted by

Kulendu Kashyap Chakraborty (190310007025)
Bishal Roy (190310007008)
Abhimanyu Kumar (190310007002)

Guided by

Dr. Minakshi Gogoi
Associate Professor & HoD
GIMT, Guwahati

Declaration

We hereby declare that this project work entitled **"Semantic Segmentation of Brain Tumor on multi-band 2D stacked-up slices using Attention-Based UNet"** was carried out by us under the guidance and supervision of Dr. Minakshi Gogoi, Associate Professor & HoD, Department of Computer Science and Engineering, Girijananda Chowdhury Institute of Management and Technology, Guwahati. This project is submitted to the Department of Computer Science and Engineering during the academic year 2022-23. The work is never produced before any authority except Assam Science and Technology University for evaluation.

Name of Project members	**Roll no**	**Signature**
Kulendu Kashyap Chakraborty	190310007025	
Bishal Roy	190310007008	
Abhimanyu Kumar	190310007002	

Certificate

This is to certify that **Kulendu Kashyap Chakraborty, Bishal Roy, Abhimanyu Kumar** students of B.Tech 4^{th} Year, 8^{th} Semester, have completed the project **"Semantic Segmentation of Brain Tumor on multi-band 2D stacked-up slices using Attention-Based UNet"** during this academic session 2022-23 under my guidance and supervision.

I approve the project for submission as required for partial fulfillment for the completion of the Bachelor of Technology Degree.

Signature of Project Guide

Dr. Minakshi Gogoi
Associate Professor & HoD,
Dept. of Computer Science and Engineering
GIMT, Guwahati.

Certificate

This is to certify that **Kulendu Kashyap Chakraborty, Bishal Roy, Abhimanyu Kumar** students of B.Tech 4^{th} Year, 8^{th} Semester, have completed the project **"Semantic Segmentation of Brain Tumor on multi-band 2D stacked-up slices using Attention-Based UNet"** during this academic session 2022-23.

I approve the project for submission as required for partial fulfillment for the completion of the Bachelor of Technology Degree.

Signature of HoD

Dr. Minakshi Gogoi
Associate Professor & HoD,
Dept. of Computer Science and Engineering
GIMT, Guwahati.

Certificate

This is to certify that **Kulendu Kashyap Chakraborty, Bishal Roy and Abhimanyu Kumar** students of B.Tech 4^{th} Year, 8^{th} semester have completed the project **"Semantic Segmentation of Brain Tumor on multi-band 2D stacked-up slices using Attention-Based UNet"** during this academic session 2022-23 under my guidance and supervision.

I approve the project for submission in partial fulfillment of the requirement for the award of the degree of **Bachelors of Technology** in **Computer Science and Engineering**.

.....	.....
(External Examiner)	(Internal Examiner)
Place:	Place:
Date:	Date

Acknowledgement

Preparing a project on a technical subject can hardly be a solitary job. There are several people who have helped us directly or indirectly in the preparation of the project. The satisfaction of completing a project successfully would remain incomplete without the mention of people whose constant encouragement and guidance have been a great source of inspiration throughout the course of this project

We would like to express my special thanks to the Project Guide **Dr. Minakshi Gogoi**, whose constant guidance and support have made the successful completion of the project. Our sincere thanks go to the Principal, **Prof. S Robert Ravi**, GIMT Guwahati and to the HOD, **Dr. Minakshi Gogoi**, Department of Computer Science and Engineering, GIMT Guwahati.

We would also like to sincerely thank all the teachers of the Department of CSE, GIMT for their encouragement of this work. Through this report, we would like to take the opportunity to express our sincere gratitude and thankfulness to all those who have helped us in making this work a success.

We also offer our heartiest thanks to our family members and friends for their constant guidance, encouragement, suggestion, and support in completing this work successfully.

Name of Project members	**Roll no**	**Signature**
Kulendu Kashyap Chakraborty	190310007025	
Bishal Roy	190310007008	
Abhimanyu Kumar	190310007002	

Abstract

A brain tumor is a cancerous and non-cancerous mass or growth of abnormal cells in the brain. It can begin elsewhere and spread to the brain. MR images of the brain provide very significant information to identify the presence of outlines concerning the brain tumor. This work employs attention-based U-Net architecture for semantic segmentation of brain tumours on multi-band 2D stacked-up slices. Semantic segmentation is a computer vision task that involves assigning a class level to every pixel in the image. An Attention-based U-Net architecture is a variant of U-Net architecture that incorporates attention mechanisms to enhance the performance of semantic segmentation tasks. It aims to selectively focus on relevant information while suppressing irrelevant or redundant information. The output of our model provides a corresponding segmented mask of the tumor.

Contents

List of Figures

List of Tables

Chapter 1

1 Introduction

1.1 Aim and Objective

Using a multi-band 3D brain scan (ideally an MRI/fMRI scan) and attention-based U-Net architecture is proposed, where we aim to isolate the tumor from the sliced layers (*derived from the volumes*) conceptually. With the information retrieved from the 2D slices from the volumes, we can build a framework to segment each pixel with a distinct feature vector.

1.2 Motivation

As a non-invasive imaging method, magnetic resonance imaging (MRI) is frequently used to find and diagnose brain disorders and track treatment. Neurologists can precisely spot irregularities in brain pictures because of the three-dimensional images produced by MRI. However, this process takes a long time and requires a lot of work. The main motivation behind this project is to create an automated system to detect and localize the area of abnormality, way ahead of time (with proper supervision). The most distinctive aspect of this project is that here 3D scans are used instead of conventional images to produce more accurate results. Patients' data can be used to analyze their stage, and supervised precautions can be taken, such as time of resection, age of onset, tumor stage, etc.

1.3 Problem Statement

Given a multi-band 3D scan of the brain (preferably MRI scans), we intend to extract all the layers as slices and aim to semantically segment the Tumor from the volumized layers using attention-based UNet architecture.

Chapter 2

2 Literature Review

2.1 Related Study (previous work)

Recent trends in detecting and super-imposing Brain Tumor through MR-Images, it's been seen that some really ground-up approaches using some techniques such as using unsupervised algorithms like template-based K-Means Clustering techniques for detecting such abnormalities and some image processing and enhancement techniques using SPC (SuperPixel Principle Component) Analysis [1]. Additionally, the utilization of process-driven U-Net architecture on MR-Images can be seen. The first step used is to transform input image data, which is further processed through various techniques—subset division, narrow object region, category brain slicing, watershed algorithm, and feature scaling was done. All these steps are implied before entering data into the U-Net Deep learning model. The U-Net Deep learning model is used to perform pixel label segmentation on the segment tumor region. The algorithm reached high-performance accuracy on the BraTS 2018 training, validation, as well as testing dataset. The proposed model achieved a dice coefficient of 0.9815, 0.9844, 0.9804, and 0.9954 on the testing dataset for sets HGG-1, HGG-2, HGG-3, and LGG-1, respectively [2].

Another survey covered the anatomy of brain tumors, publicly available datasets, enhancement techniques, segmentation, feature extraction, classification, deep learning, transfer learning, and quantum machine learning for brain tumor analysis. Finally, this survey provides all important literature for the detection of brain tumors with their advantages, limitations, developments, and future trends [3]. Additionally, it has been observed that the system might use image processing and machine learning algorithms to the image to produce the desired results. The fact that the edges of MRI pictures are not sharp in the early stages of brain tumors is one of the problems with them. Therefore, image segmentation is performed on MRI images to find edges. Other filtering methods could be used to increase the system's accuracy and effectiveness [4]. We also noticed the development of an automatic segmentation method based on CNNs (convolution neural networks). Segmentation and categorization are achieved by utilizing this single technique. CNN (a machine learning technique) is derived from neural networks and has layered bases for different outputs [5].

We've also seen a suggested project involving machine learning approaches that are broken up into three sections: preprocessing of brain MRI images, texture feature extraction using a Gray Level Co-occurrence Matrix (GLCM), and classification using a machine learning method [6]. A hybrid ensemble technique based on the majority voting method and using Random Forest (RF), K-Nearest Neighbor, and Decision Tree (DT) (KNNRF-DT) was presented in another suggested work. In order to categorize brain tumors as benign or malignant, the area of the tumor region is calculated. Otsu's Threshold technique is initially used for segmentation. Thirteen characteristics are extracted for classification utilizing the Stationary Wavelet Transform (SWT), Principle Component Analysis (PCA), and Gray Level Co-occurrence Matrix (GLCM). Based on the Majority Voting approach, a hybrid ensemble classifier (KNN-RFDT) performs the classification [7]. We have also read an article that discusses the various phases of image

processing and provides an overview of related literature by reviewing a number of studies. This article summarizes the technologies that can be used to forecast brain tumors [8]. Another study focuses on the application of TensorFlow to MRI-based brain cancer detection. We constructed a convolutional neural network with five layers in TensorFlow. In this dataset, 1800 MRIs were used, of which 900 were malignant and 900 were not. In 35 epochs, it was discovered that the validation accuracy was 98.6% and the training accuracy was 99%. This program is still being worked on. Surgeons and radiologists can utilize the technology as a second opinion to quickly and effectively find brain tumors [9].

We also observed hybrid deep learning models using machine learning methods in a novel fashion (e.g., AlexNet, AlexNet+SVM, ResNet-18, and ResNet-18+SVM). With the help of the average and Laplacian filters, images were enhanced. To extract deep and discriminatory information, improved photos were fed into deep learning models. CNN classifiers, which are SoftMax, and machine learning classifiers known as SVM algorithms were used to diagnose deep features. MRI scans of brain tumors were successfully identified using all proposed systems, with little variation in accuracy between models. The computational costs for training the dataset differ significantly from one another [10].

Additionally, we have seen methods for grading (classifying) brain tumours into three categories using CNN and 3064 T1 weighted contrast-enhanced brain MR images of varied sizes (cropped and uncropped) (Glioma, Meningioma, and Pituitary Tumour) [11]. Another technique is to build a big dataset of synthetic MRI pictures that mimic the typical pattern of brain MRI images using a small class-unbalanced acquired dataset. The acquired dataset is then used to train a deep model for detection and classification [12]. In the context of machine learning and the deep learning paradigm, one approach has also emphasized the connection between brain cancer and other brain disorders like stroke, Alzheimer's, Parkinson's, and Wilson's disease [13]. We have also seen another method that improved classification accuracy with Flair (>99%), T2 (>98%), T1C (>97%), and clinical pictures (>98%) by combining VGG19 with SVM-RBF [14]. Given how challenging it is to identify brain tumours using computer vision algorithms, there is also a technique that combines CNN and SVM to provide a classifier for brain tumours [15]. We have also seen a method that uses VGG16 and ResNet101 to do detection via transfer learning. The goal of transfer learning in machine learning is to apply the knowledge acquired while resolving one problem to another, unrelated problem [16]. Another method demonstrates how doctors and radiologists can identify and discover the inner human anatomy components without the need for surgery by using various radiology techniques. And segmentation analysis of brain tumour MR images is carried out utilising such methods [17].

Hyper-Layer Convolutional Neural Networks (HL-CNN) and Hyper-Heuristic Extreme Learning Machines are introduced in this technique (HH-ELM). It introduces the HL-CNN for feature extraction. The hyper-layer technique is a masking method that does not just take into account the features at the top layer but also masks the features of the specified layers. To reduce the number of pointless features in the system, the best features are chosen using HL-CNN validation utilizing a straightforward correlation-based selection method. The HH-ELM is introduced in the last stage to categorize the tumour images and determine the various types of tumours [18]. We have also seen an approach where to extract reliable features and learn the structure of MR pictures in its convolutional layers, a deep neural network is first pre-trained as a

discriminator in a generative adversarial network (GAN). This is done on several datasets of MR images. The entire deep network is then trained as a classifier to differentiate between three tumour classifications after the fully connected layers are replaced. When compared to cutting-edge techniques, 5-Fold cross-validation produces the highest accuracy when used to assess the performance of an overall design [19]. This study uses support vector machine classifiers to use spectroscopy to determine the type of tissue in human brain samples. In order to collect infrared spectroscopic traces in the wavenumber range between 1200 and 3500 cm1, two separate spectrometers were used. The outcomes show that the created algorithm is strong enough to categorize the infrared data from human brain tissue at three different levels of discrimination[20].

Attention-based UNet architectures have gained popularity in image segmentation tasks, [21] uses 2D UNet architecture in which two techniques have been employed. The first technique is an attention mechanism, which is adopted after concatenation of low level features and high level features that prevent confusion of the model by weighting each of channels adaptively. The second technique is Multi-view Fusion that provides 3D contextual information of input images by extracting 2D slices from both Axial and Coronal views using a 2D model. The paper utilizes the BRATS 2017 and BraTs 2018 datasets for experiment. Similarly [22] demonstrate UNet Based Global Attention Mechanism (GAU-Net) that combines channel attention and spatial attention mechanism, and integrates different convolutional layers for brain tumor segmentation.This experiment is performed on BraTs 2018 that provides mIoU 0.75 with only 5.4% of UNet parameters. [23] has developed a Multi-Threshold model based on Attention UNet(MTAU) for Identification of various regions of tumour in MRI and proposed a multi-path segmentation by building three separate models for the different regions of interest. The proposed model utilizes BraTs 2020 datasets that gives mean Dice Coefficients of 0.57,0.73 and 0.61 for enhancing tumor, whole tumor and tumor core respectively.

Chapter 3

3 Background Study

3.1 Neural Network Architectures for Bio-Medical Image Processing

Innovation in neural networks often happens through architectural change. In fact, architectural change is arguably what popularized deep neural networks (DNNs). The convolutional variety of DNNs had their first mainstream success in 1989 for classification in a now-antiquated model called the LeNet. In 2012, the AlexNet was introduced for a more difficult dataset, but the blocks which composed the AlexNet were mostly the same as the blocks which composed the LeNet. The innovation of the AlexNet was mostly in how the blocks were situated with respect to one another and the number of blocks used—in other words, the architecture1.

If architecture plays such an important role in performance, it is natural to ask how to choose architecture. In computer vision, there has been a slew of papers that introduce both radically different architectures and incremental improvements. In medical image segmentation, however, the architecture often seems to default to the U-Net.

The U-Net is a simple-to-implement DNN architecture that has been wildly successful in medical imaging; the paper that introduces the U-Net, published in 2015, is the most cited paper at the prestigious medical imaging conference MICCAI. Over the past five years, however, there have been significant improvements to architectures for segmentation tasks, and it is no longer safe to assume that the U-Net will provide near state-of-the-art performance.

3.2 Deep Learning for Semantic Segmentation

The aim of Semantic Segmentation is to classify each pixel of an image. Image segmentation is a computer vision task that involves labelling various regions of the image into objects that are present in it.

Convolutional Neural Networks (CNNs) have been pretty common in the field of Deep Learning nowadays. CNN's have been used for computer vision tasks like image classification, object detection, image generation etc. Like all other computer vision tasks, deep learning has been known to outperform previously existing image segmentation approaches.

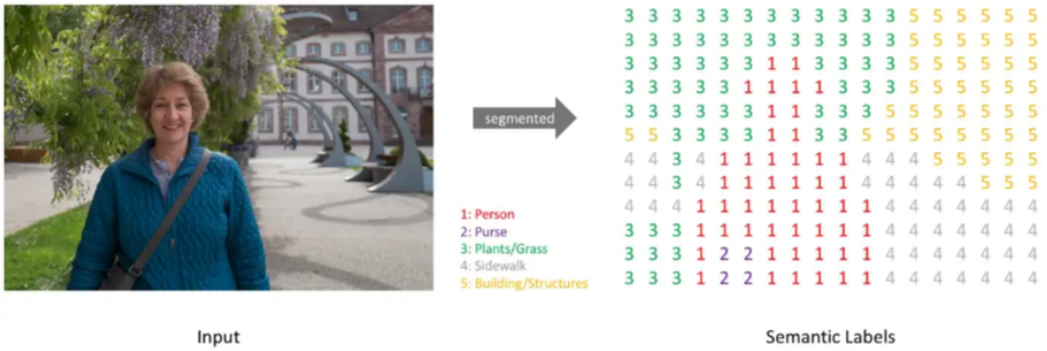


Figure 1: Class labels for input image *(semantically labelled classes for each class of pixels)*

3.2.1 CNNs for Semantic Segmentation

Like other computer vision tasks, using a CNN for semantic segmentation would be the obvious choice. When using CNN for semantic segmentation, the output would be an image of the same resolution as the input, unlike a fixed-length vector in the case of image classification. The general architecture of models contains a series of convolutional layers along with pooling or strode convolutional layers for downsampling. To improve the model, non-linear activations and batch normalization layers are also used. The initial layers in a convolutional neural network learn low-level features like lines, edges, colours, etc. and the deeper layers learn high-level features like faces or objects, etc.

Shallower convolutional layers contain more information about smaller regions of the image. This is because when dealing with high-dimensional inputs like an image, it is impractical to connect each neuron to all neurons in the previous volume. Instead, we connect each neuron to only a local region of the input volume. The spatial extent of this connectivity is called the receptive field of the neuron. Thus, as we add more layers, the size of the image keeps on decreasing and the number of channels keeps on increasing. The downsampling is done by the pooling layers.

For image classification, we reduce the size of the input image using a recurring set of convolutional and pooling layers, and finally when the size of the image is small enough we flatten out the feature map and feed it into a fully-connected layer for classification purposes. So essentially, we are mapping the input image to a fixed-size vector.

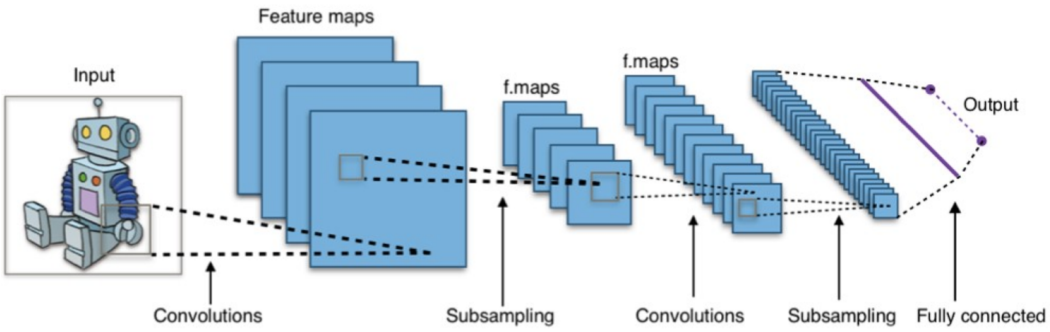


Figure 2: CNN for image classification[2]

But flattening out the feature maps leads to a loss of spatial information, which is essential to the process of semantic segmentation. To retain the spatial information, no fully connected layers are used in the network. Convolutional layers are coupled with downsampling layers to produce a low-resolution spatial tensor. This tensor contains high-level information about the input contained in its various channels. In most implementations, this tensor has the lowest resolution with the largest amount of channels. Now that we have obtained this low-resolution tensor, we somehow have to increase its resolution up to the original image to achieve the task of semantic segmentation. We feed this low-resolution feature map to upsampling layers followed by more convolution layers to create higher-resolution feature maps. As we increase the resolution we simultaneously decrease the number of channels in the feature maps. This kind of architecture is known as encoder-decoder architecture. The downsampling phase is known as the encoder and the upsampling phase is called the decoder.

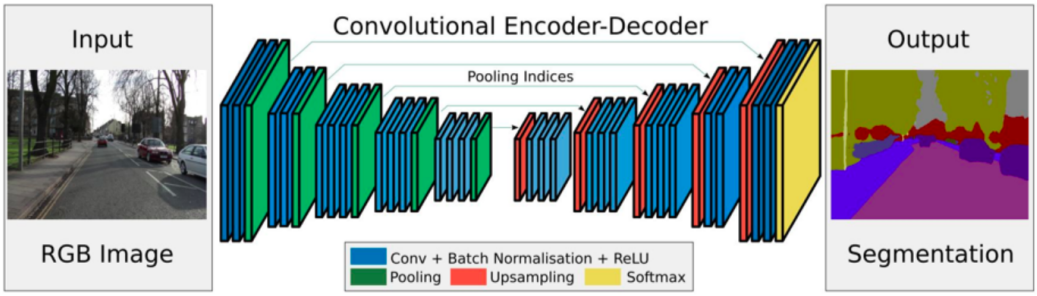


Figure 3: Encoder-Decoder based Architecture[2]

3.2.2 Attention-based Networks for computer vision tasks

Attention-based networks have emerged as powerful tools for various computer vision tasks, including object detection, image classification, image segmentation, and more. These networks utilize the concept of attention mechanisms, inspired by human visual attention, to focus on relevant image regions while processing visual data. Attention-based networks have also found significant applications in the medical field, revolutionizing medical image analysis and aiding in diagnosis, treatment planning, and disease prediction.

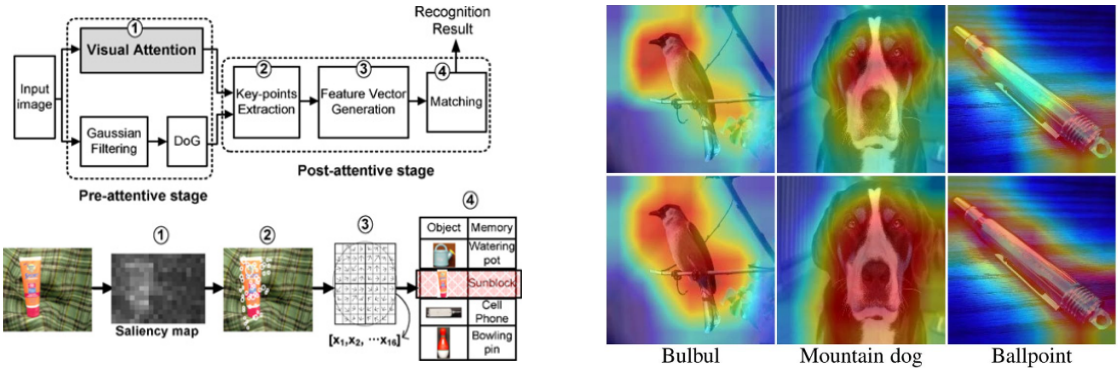
In computer vision tasks, attention mechanisms enable the network to selectively attend to specific regions of an image, allowing it to assign different weights or importance to different parts of the input. This selective focusing improves the network's ability to extract meaningful features from images and capture fine-grained details. By adaptively learning where to focus their attention, these networks can allocate computational resources more efficiently, leading to improved performance and reduced computational complexity.

One popular attention mechanism used in computer vision tasks is called "self-attention" or "transformer attention." Self-attention computes attention weights for all pairs of positions in an input sequence and combines them to generate contextualized representations. This mechanism allows the network to capture long-range dependencies and model relationships between different elements in the input, which is particularly useful for tasks that require understanding the global context of an image.

In the medical domain, attention-based networks have made significant contributions to various applications, such as medical image classification, segmentation, and anomaly detection. For instance, in medical image classification, attention mechanisms can help the network focus on important regions within an image that contains diagnostically relevant features. This enables more accurate and interpretable predictions, improving the performance of automated diagnosis systems. By adaptively attending to relevant anatomical regions, these networks can generate precise segmentation masks, facilitating the measurement of organ volumes, tumor localization, and treatment planning.

Attention mechanisms have also been utilized for anomaly detection in medical imaging. By attending to abnormal regions or regions that deviate from normal patterns, attention-based networks can effectively identify anomalies in medical images, aiding in the early detection of diseases and assisting radiologists in their assessments.

Furthermore, attention-based networks have been combined with other techniques, such as transfer learning and multi-modal fusion, to leverage complementary information from different medical imaging modalities (e.g., MRI, CT, PET) and improve the accuracy of disease prediction, prognosis estimation, and treatment response assessment. Overall, attention-based networks have demonstrated their potential in computer vision tasks and have found wide applications in the medical field. By incorporating attention mechanisms, these networks can enhance the performance, interpretability, and efficiency of medical image analysis systems, contributing to improved patient care and medical decision-making.



(a) Attention in a CNN/MLP

(b) Output of the attention-ed images

Figure 4: Visualization of the Attention mechanism

Chapter 4

4 Proposed Methodology

4.1 Overview of the work

This study and work of semantically segmenting Brain Tumors from MR-Images comprehending with multi-band channels on 3D volumes consist of the following steps:

- **Step 1:** Understanding the Dataset (*BraTS'20 Challenge Data*), with all the medical arrangements of the terminologies.
- **Step 2:** Generation and Preprocessing of the Data.
- **Step 3:** Defining of the 3D U-Net Architecture with the Attention mechanism.
- **Step 4:** Training the Segmentation Model (*U-Net*).
- **Step 5:** Performance analysis of the models trained and generated outputs.
- **Step 6:** Evaluation with the Benchmark parameters and comparing the Mask.

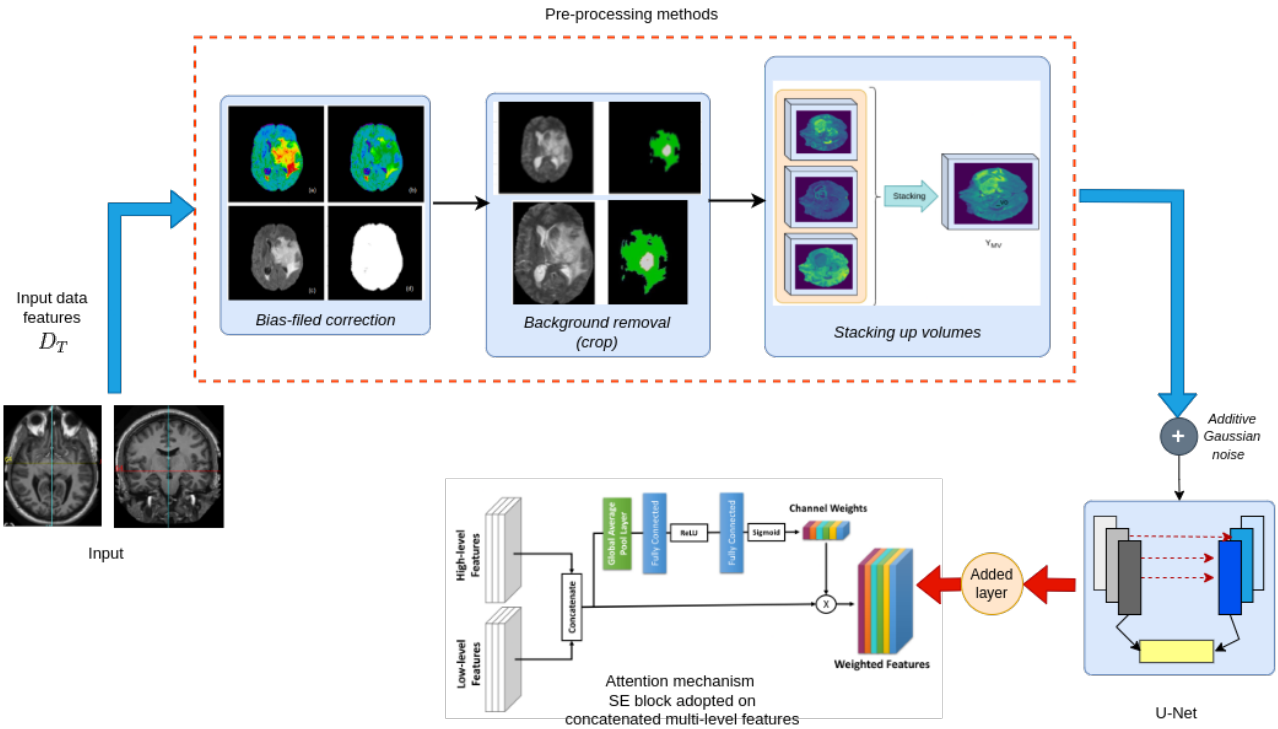


Figure 5: Proposed Methodology of the work

4.2 Data generation and pre-processing

4.2.1 Understanding of the Dataset

Ample multi-institutional routine clinically-acquired pre-operative multimodal MRI scans of glioblastoma (GBM/HGG) and lower grade glioma (LGG), with pathologically confirmed diagnosis and available OS, are provided as the training, validation, and testing data for the BraTS'20 challenge. Specifically, the datasets used in the challenge have been updated, since BraTS'19, with more routine clinically-acquired 3T multimodal MRI scans, with accompanying ground truth labels by expert board-certified neuroradiologists.

4.2.2 Imaging Data Description

All BraTS multimodal scans are available as NIfTI files (.nii.gz) and described as:

- native (T1)
- post-contrast T1-weighted (T1Gd)
- T2-weighted (T2)
- T2 Fluid Attenuated Inversion Recovery (T2-FLAIR) volumes,

and were acquired with different clinical protocols and various scanners from multiple (n=19) institutions, mentioned as data contributors here.

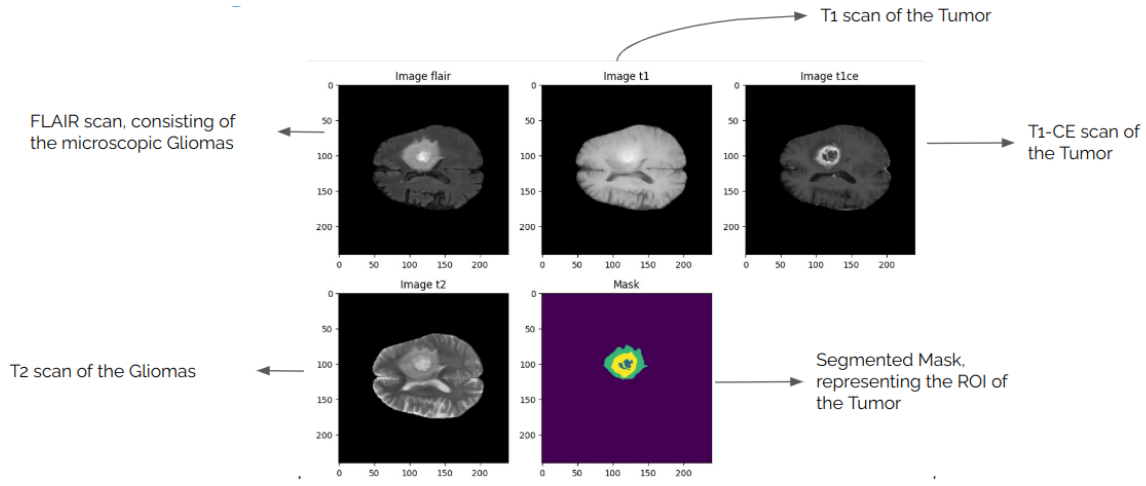


Figure 6: Scans of all the modalities of a MRI image, where the Mask is the segmented tumor

All the imaging datasets have been segmented manually, by one to four raters, following the same annotation protocol, and their annotations were approved by experienced neuro-radiologists. Annotations comprise the GD-enhancing tumor (ET — label 4), the peritumoral edema (ED — label 2), and the necrotic and non-enhancing tumor core (NCR/NET — label 1), as described both in the BraTS 2012-2013 TMI paper and in the latest BraTS summarizing paper. The provided data are distributed after their pre-processing, i.e., co-registered to the same anatomical template, interpolated to the same resolution ($1mm^3$), and skull-stripped.

4.2.3 Comparison with the Previous BraTS datasets

The BraTS data provided since BraTS'17 differs significantly from the data provided during the previous BraTS challenges (i.e., 2016 and backward). The only data that have been previously used and are utilized again (during BraTS'17-'20) are the images and annotations of BraTS'12-'13, which have been manually annotated by clinical experts in the past. The data used during BraTS'14-'16 (from TCIA) have been discarded, as they described a mixture of pre-and post-operative scans and their ground truth labels have been annotated by the fusion of segmentation results from algorithms that ranked highly during BraTS'12 and '13. For BraTS'17, expert neuroradiologists have radiologically assessed the complete original TCIA glioma collections (TCGA-GBM, n=262 and TCGA-LGG, n=199) and categorized each scan as pre- or post-operative. Subsequently, all the pre-operative TCIA scans (135 GBM and 108 LGG) were annotated by experts for the various glioma sub-regions and included in this year's BraTS datasets.

The BraTS'2020 data provides the naming convention and direct filename mapping between the data of BraTS'20-17, and the TCGA-GBM and TCGA-LGG collections, available through The Cancer Imaging Archive (TCIA) to further facilitate research beyond the directly BraTS-related tasks.

4.3 Data acquisition and preprocessing

4.3.1 Data generation

The data comes with some processing applied and on top of that these cannot be synthetically augmented because of the sensitiveness and precision of the localized Tumors. So, in order to be more precise, the following pre-processing techniques have been imposed:

1. **Cropping the Region of 'Abnormality', and focusing more on the ROI:**

   The original (*raw*) images contain a lot of pixel information and are kind of generalized for any segmentation and declaration tasks. The volumized images are captured in such a way that apart from the ROI (or the area of abnormality) there are a lot of redundant/null information pixels, which will not be beneficial for extracting any kind of information. So "Cropping" the area containing only the tumors was the only way to get rid of all the null pixels so that while operating on them, maximum information can be retrieved.

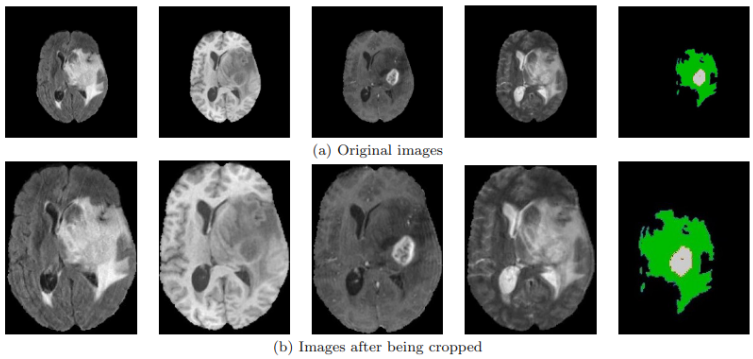


Figure 7: Cropped image from the Data. The columns from left to right are FLAIR sequences, T1 sequences, T1-CE sequences, T2 sequences and their corresponding masks

2. **Stacking the 'Combined Volumes':**

There are four classes of differentiating Tumors with each providing different information about one's surroundings and core values, resulting in distinct pixel-neighborhoods. Since all the other volumes provide abstract properties, a mega-volume is generated by stacking and combining the FLAIR, T1, and T1-CE volumes with the same resolution as the other sub-volumes and having all the heterogeneous properties in a multi-volume single scan.

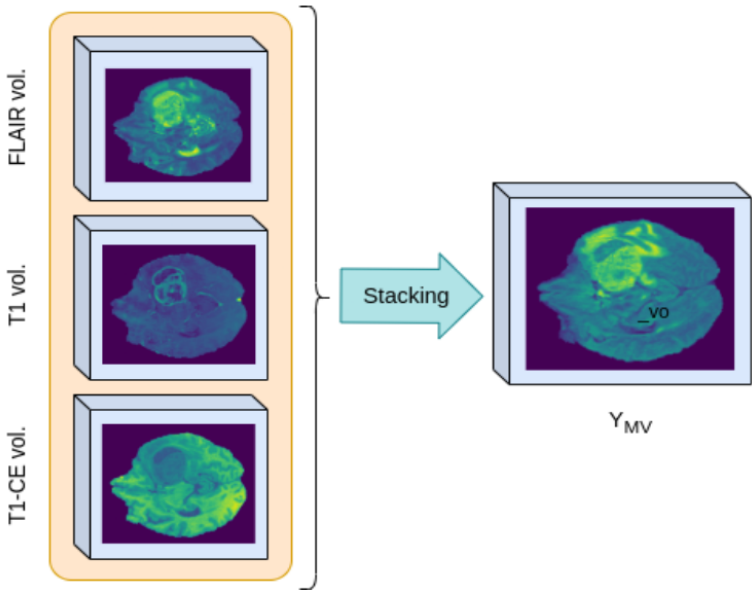


Figure 8: Stacking combined volumes in a multi-volume single scan.

3. **Bias-field Correction**

Sometimes the MRI scans generated by the scanner acquire a low-frequency intensity gradient and extremely smooth signal across the images making some parts of the images brighter than others which is known as the bias field or gain field. These signals are

barely perceptible to the human eye but they are highly sensitive to processing algorithms, Therefore a pre-processing step is required to correct the bias field signals before putting corrupted MRI scans into models or processing algorithms. In this approach, we are using the N4 bias field correction algorithm for correcting low-frequency intensity non-uniformity appearing in MRI images. Among the multiple implementations of N4 bias field correction, We have used the SimpleITK(SITK) library which is most practical and easy to use.

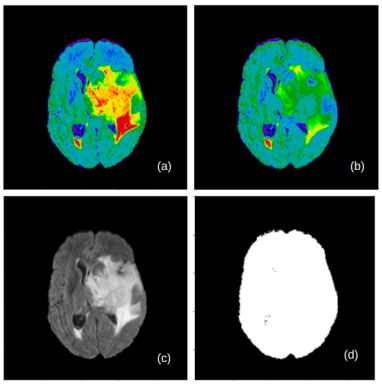


Figure 9: Bias corrected output. (a) Original image, (b) bias field corrected, (c) greyscale image, (d) Transformed Threshold Mask

4.4 Pipeline of the work

4.4.1 Explanation of the U-Net Architecture

UNet [25], which evolved from the traditional convolutional neural network, was first designed and applied in 2015 to process biomedical images. As a general convolutional neural network focuses its task on image classification, where input is an image and output is one label, but in biomedical cases, it requires us not only to distinguish whether there is a disease but also to localize the area of abnormality.

UNet is dedicated to solving this problem. The reason it is able to localize and distinguish borders is by doing classification on every pixel, so the input and output share the same size.

4.4.2 Overview of the Network

At a very first glance, the architecture has a “U" shape. The architecture is symmetric and consists of two major parts: the left part is called the *“Contracting path"*, which is constituted by the general convolutional process; the right part is the *“Expansive path"*, which is constituted by transposed 2D Convolutional Layers. One important modification in U-Net is that there are a large number of feature channels in the upsampling part, which allow the network to propagate context information to higher resolution layers. As a consequence, the expansive path is more or less symmetric to the contracting part. The network only uses the valid part of each convolution without any fully connected layers. To predict the pixels in the border region of the image, the missing context is extrapolated by mirroring the input image. This tiling strategy is important to apply the network to large images since otherwise the resolution would be limited by the GPU memory.

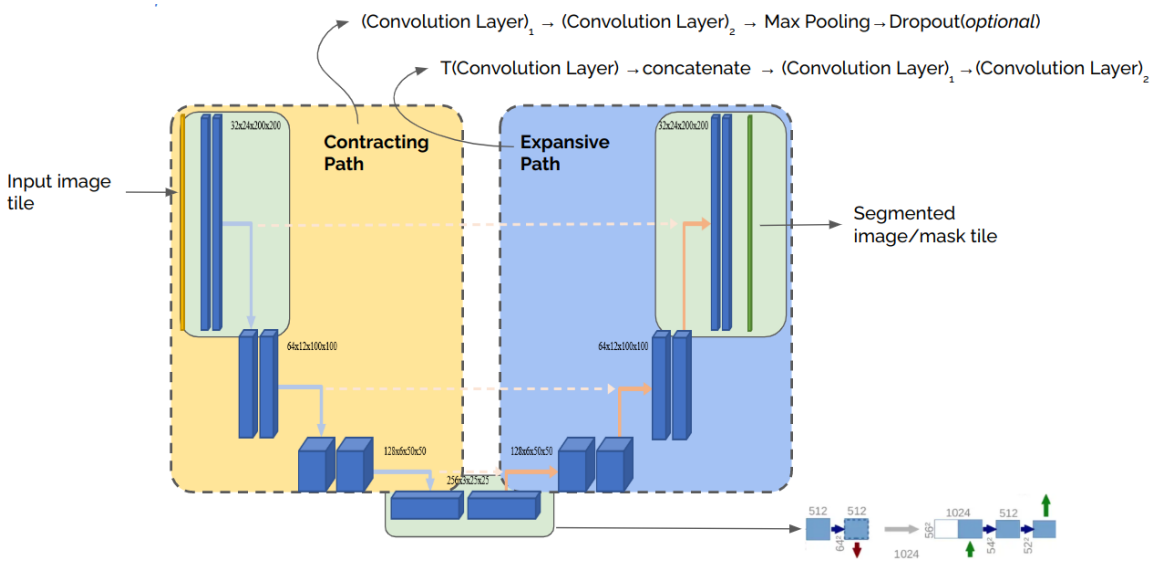


Figure 10: U-Net Architecture [2]

4.4.3 Contracting path

It consists of the repeated application of two 3x3 convolutions. Each convolution is followed by a ReLU and batch normalization. Then a 2x2 max pooling operation is applied to reduce the spatial dimensions. Again, at each downsampling step, we double the number of feature channels, while we cut in half the spatial dimensions. The following is the flow of the operations:

$$(ConvolutionLayer)_1 \rightarrow (ConvolutionLayer)_2 \rightarrow MaxPooling \rightarrow Dropout(optional)$$

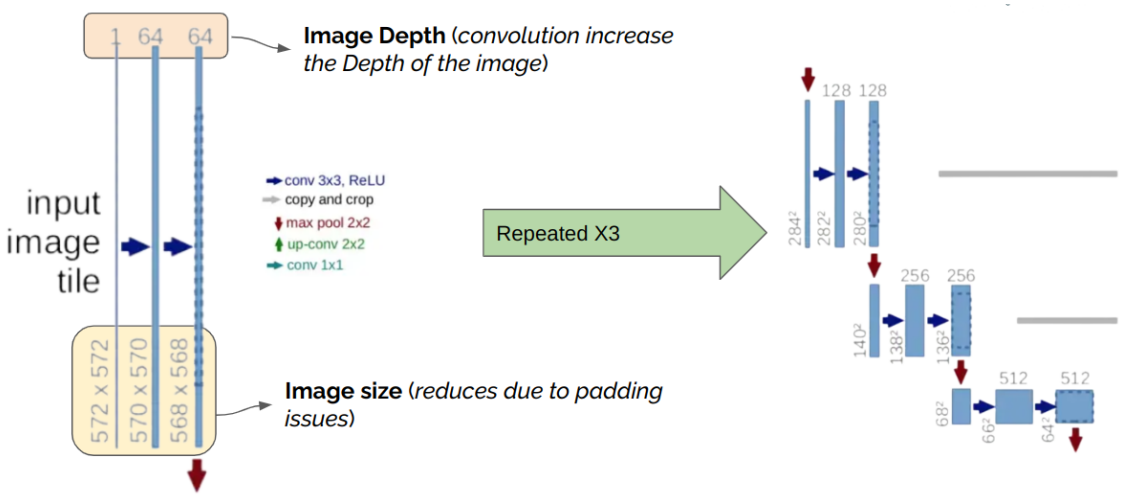


Figure 11: Contracting Path [2]

4.4.4 Expansive path

Transposed convolution is an upsampling technic that expands the size of images. Basically, it does some padding on the original image followed by a convolution operation.

After the transposed convolution, the image is upsized from 28x28x1024 $\rightarrow$ 56x56x512, and then, the image is concatenated with the corresponding image from the contracting path and together make an image of size 56x56x1024. The reason here is to combine the information from the previous layers in order to get a more precise prediction.

$$(TransposedConvolutionLayer)_1 \rightarrow concatenate \rightarrow (ConvolutionLayer)_1 \rightarrow (ConvolutionLayer)_1$$

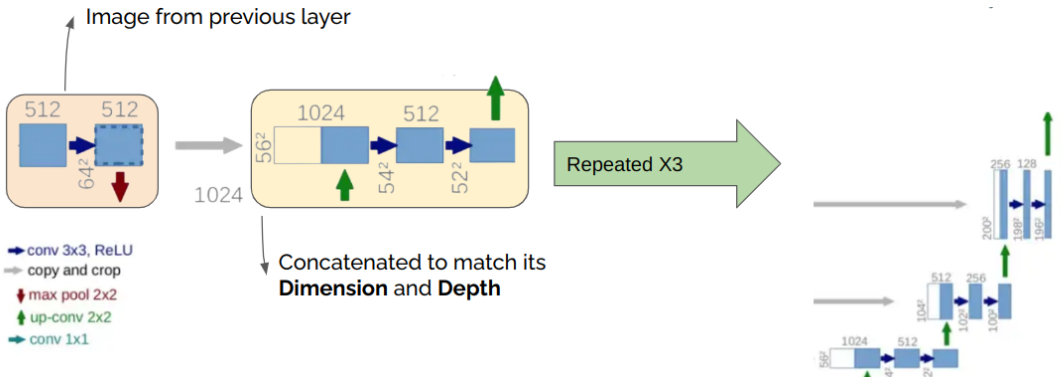


Figure 12: Expansive Path [2]

4.4.5 Attention mechanism

The attention mechanism is a crucial component in the U-Net architecture due to its ability to enhance the model's performance in capturing intricate spatial dependencies and focusing on relevant information. The U-Net is a popular convolutional neural network (CNN) architecture designed for image segmentation tasks, where the objective is to assign a class label to each pixel in an image. In the U-Net architecture, the network consists of an encoding path and a decoding path. Convolutional and pooling layers are used in the encoding pipeline to capture high-level characteristics and shrink the spatial dimensions of the input image. The decoding method then applies upsampling and concatenation procedures to recover the finer details of the segmented regions while reconstructing the spatial information.

The U-Net's attention mechanism is crucial to the decoding path. The attention mechanism enables the network to selectively focus on informative regions and inhibit irrelevant or noisy areas as the decoding path gradually recovers the spatial information. The segmentation accuracy is increased as a result of the model's ability to dedicate more resources to regions of interest through selective attention.

One common approach to implementing the attention mechanism in U-Net is through the use of skip connections. These connections establish links between corresponding encoding and decoding layers, facilitating the integration of local and global context information. By combining features from different resolution levels, the attention mechanism helps the U-Net to capture both fine-grained details and high-level context simultaneously. Additionally, U-Net's attention system helps to address concerns about class inequality. It frequently happens in picture segmentation

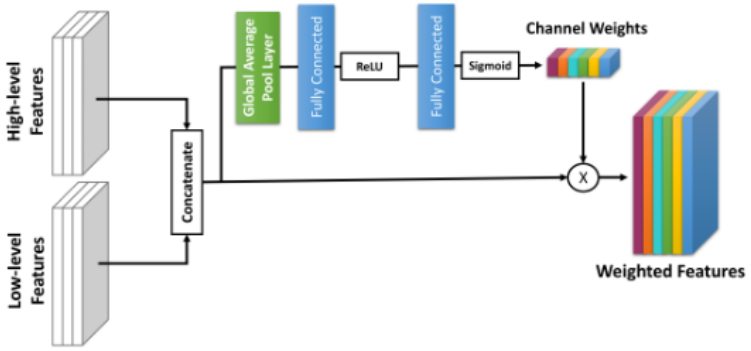


Figure 13: Attention architecture

tasks that some classes or regions of interest are noticeably smaller in size than others. Without consideration, the model might have trouble correctly segmenting these smaller areas. The U-Net's segmentation performance is enhanced by the attention mechanism, which enables the U-Net to adaptively alter its focus and pay more attention to such difficult regions.

In summary, the U-Net attention mechanism is essential for accurately detecting spatial relationships, attending just to pertinent information, fusing local and global context, and managing class imbalance. The U-Net design improves its ability to reliably segment images by adding attention, and it has demonstrated promising results in a number of computer vision and medical imaging applications.

4.4.6 Squeeze and Excitation (SE) Block

As we know CNN uses its convolution filter to extract hierarchical information from the images. The Lower layers detect lines, edges e.t.c while the upper layers can detect faces, text, or complex geometrical shapes. The convolution operator generates a feature map with the different number of channels, where it assigns the weight of each channel equally. It means that every single channel is equally important and this may not be the best way. The Squeeze and Excitation Block assign the weight of each channel adaptively based on the context of each channel.

The Squeeze and Excitation Block (SE Block) [24] is an architectural unit designed to improve the representational power of a network by enabling it to perform dynamic channel-wise feature re-calibration at almost no computation cost. It can be easily added to the existing architecture. In 2018, The SE blocks were used in the ImageNet competition which improved the result obtained from last year by 2.5%.

Let's understand the working of the SE block. The SE block takes a convolutional block ($CXHXW$) as input and it is passed to squeeze operation to get a global understanding of each channel by squeezing the feature map to a single numeric value using average pooling. This step results in a vector of size n(where n is determined by the number of channels in input). After that, it passes to the excitation operation where this vector is fed to a two-layer feed-forward network. The first dense layer is followed by a ReLU that adds non-linearity and reduces the

complexity of the output channel by a ratio and the other dense layer is followed by a sigmoid that gives each channel a smooth gating function. Finally, the network outputs a vector of the same size (n). These n values can be used as weights on the original features maps or convolutional block, scaling each channel based on its importance.

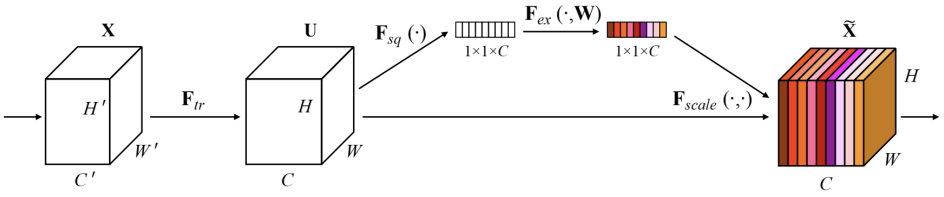


Figure 14: Squeeze and Excitation Block [24]

4.4.7 Gaussian noise

Gaussian Noise, also known as adaptive White Gaussian noise(AWGN), is a type of random noise that is generated by adding random values that are normally distributed with a mean of zero and a standard variation to the input data. The normal distribution is also known as the Gaussian distribution. It is commonly used in machine learning or Deep learning as a way to introduce randomness into input data or models. The main goal of adding Gaussian noise is to improve the model's performance and robustness without making it too difficult for the model to learn from the data.

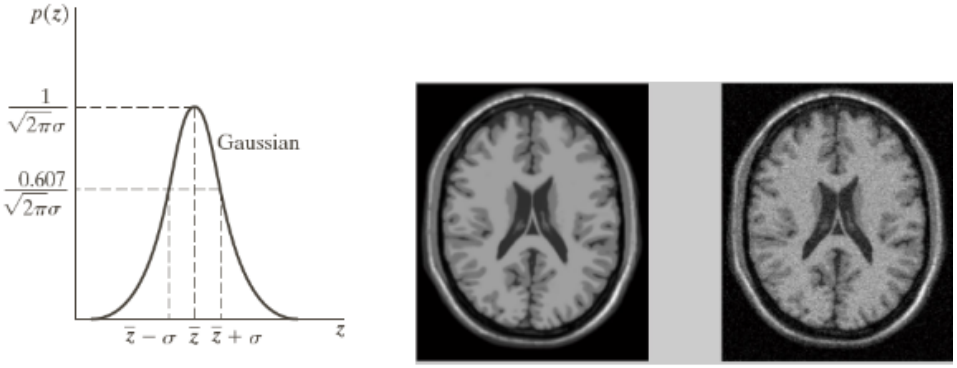


Figure 15: Gaussian noise distribution

Adding Gaussian noise as the first layer of the model serves the following purposes:

1. **Regularization**: By adding random noise to the input data, Gaussian noise acts as a regularizer that helps to prevent overfitting and encourages models to learn more robust and generalized features.

2. **Data Augmentation**: Gaussian noise can be used as a form of data augmentation. By adding noise to input data, the model is forced to learn the features that are robust to small variations of the original data. This improves the model's ability to generalize and handle variations in input during training.

3. **Robustness to perturbations**: Models that are trained with Gaussian noise as the first layer tend to be more robust to small perturbations in the input. It is beneficial where data are noisy and prone to small errors, as the model learns to be less sensitive to such variation.

We can add Gaussian noise before the first layer of a network or we can use different deep learning frameworks such as TensorFlow or PyTorch without implementing them from scratch. The noise layer will introduce random values drawn from a Gaussian distribution to the input data, which then propagates through the subsequent layers of the model during training.

4.5 Training the Segmentation Model (DNN)

During the training of the architecture, the implementation of the model training consists of two main steps. Specifically, the first step is to import the raw contracting path, with all the layers (consisting of the convolutional, and ReLU (1), as well as the pooling layers). And the second step was to define the expansive path on top of it, to train the full model according to our segmentation task.

$$f(x) = max(0, x) \tag{1}$$

Sample Configuration for the Model

- Batch size = 2
- Verbose = 1
- Epochs = 50
- Training sample size = 248 samples
- Validation sample size = 96 samples

Parameter and Hyper-Parameter configuration

- Learning Rate (α) = 0.0001
- Steps per Epochs = 50
- Validation steps per Epochs = 50
- Optimizer = Softmax 2
- Padding = Same as the resolution of the previous layer.

$$\sigma(\overrightarrow{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{2}$$

4.5.1 Loss functions

Loss functions:

- Dice Loss: $L_{dice} = \frac{2*\sum p_{true}*p_{pred}}{\sum p_{true}^2+\sum p_{pred}^2+\epsilon}$
- Focal Loss: $L_{focal} = -\sum_{i=1}^{n} \alpha_i(i-p_i)^\gamma log_b(p_i)$

$\boxed{\text{Total Loss: } L_{dice} + L_{focal}}$

Code snippet of the weighted losses

```
# defining the weighted losses
import segmentation_models_3D as sm

wt0, wt1, wt2, wt3 = 0.25,0.25,0.25,0.25

dice_loss = sm.losses.DiceLoss(class_weights=np.array([wt0, wt1, wt2, wt3]))
focal_loss = sm.losses.CategoricalFocalLoss()
total_loss = dice_loss + (1 * focal_loss)
```

Now, with all the above-listed configurations we have initialized the model and trained the model. On training it on the training set, the recorded accuracy so far is 97.67%.

Table 1: Detailed architecture of the u-net model

Layer (type)	**Output Shape**	**Parameters**	**Connected to**
input_1 (InputLayer)	[(None, 128, 128, 1, 28, 3)]	0	[ ]
conv3d (Conv3D)	(None, 128, 128, 12, 8, 16)	1312	['input_1[0][0]']
dropout (Dropout)	(None, 128, 128, 12, 8 16)	0	['conv3d[0][0]']
conv3d_1 (Conv3D)	(None, 128, 128, 12, 8, 16)	6928	['dropout[0][0]']
max_pooling3d (MaxPooling3D)	(None, 64, 64, 64, 16)	0	['conv3d_1[0][0]']
conv3d_2 (Conv3D)	(None, 64, 64, 64,32)	13856	['max_pooling3d[0][0]']
dropout_1 (Dropout)	(None, 64, 64, 64,32)	0	['conv3d_2[0][0]']
conv3d_3 (Conv3D)	(None, 64, 64, 64,32)	27680	['dropout_1[0][0]']
max_pooling3d_1 (MaxPooling3D)	(None, 32, 32, 32,32)	0	['conv3d_3[0][0]']
conv3d_4 (Conv3D)	(None, 32, 32, 32,64)	55360	['max_pooling3d_1[0][0]']
dropout_2 (Dropout)	(None, 32, 32, 32,64)	0	['conv3d_4[0][0]']
conv3d_5 (Conv3D)	(None, 32, 32, 32,64)	110656	['dropout_2[0][0]']
max_pooling3d_2 (MaxPooling3D)	(None, 16, 16, 16, 64)	0	['conv3d_5[0][0]']
conv3d_6 (Conv3D)	(None, 16, 16, 16,128)	221312	['max_pooling3d_2[0][0]']
dropout_3 (Dropout)	(None, 16, 16, 16, 128)	0	['conv3d_6[0][0]']
conv3d_7 (Conv3D)	(None, 16, 16, 16,128)	442496	['dropout_3[0][0]']
max_pooling3d_3 (MaxPooling3D)	(None, 8, 8, 8, 128)	0	['conv3d_7[0][0]']
conv3d_8 (Conv3D)	(None, 8, 8, 8, 256)	884992	['max_pooling3d_3[0][0]']
dropout_4 (Dropout)	(None, 8, 8, 8, 256)	0	['conv3d_8[0][0]']
conv3d_9 (Conv3D)	(None, 8, 8, 8, 256)	1769728	['dropout_4[0][0]']
conv3d_transpose (Conv3DTranspose)	(None, 16, 16, 16, 128)	262272	['conv3d_9[0][0]']
concatenate (Concatenate)	(None, 16, 16, 16, 256)	0	['conv3d_transpose[0][0]','conv3d_7[0][0]']
conv3d_10 (Conv3D)	(None, 16, 16, 16,128)	884864	['concatenate[0][0]']
dropout_5 (Dropout)	(None, 16, 16, 16,128)	0	['conv3d_10[0][0]']
conv3d_11 (Conv3D)	(None, 16, 16, 16,128)	442496	['dropout_5[0][0]']
conv3d_transpose_1 (Conv3DTranspose)	(None, 32, 32, 32,64)	65600	['conv3d_11[0][0]']
concatenate_1 (Concatenate)	(None, 32, 32, 32, 128)	0	['conv3d_transpose_1[0][0]','conv3d_5[0][0]']
conv3d_12 (Conv3D)	(None, 32, 32, 32, 64)	221248	['concatenate_1[0][0]']
dropout_6 (Dropout)	(None, 32, 32, 32,64)	0	['conv3d_12[0][0]']
conv3d_13 (Conv3D)	(None, 32, 32, 32, 64)	110656 z	['dropout_6[0][0]']
conv3d_transpose_2 (Conv3DTranspose)	(None, 64, 64, 64, 32)	16416	['conv3d_13[0][0]']
concatenate_2 (Concatenate)	(None, 64, 64, 64, 64)	0	['conv3d_transpose_2[0][0]', 'conv3d_3[0][0]']
conv3d_14 (Conv3D)	(None, 64, 64, 64,32)	55328	['concatenate_2[0][0]']
dropout_7 (Dropout)	(None, 64, 64, 64,32)	0	['conv3d_14[0][0]']
conv3d_15 (Conv3D)	(None, 64, 64, 64, 32)	27680	['dropout_7[0][0]']
conv3d_transpose_3 (Conv3DTranspose)	(None, 128, 128, 12, 8, 16)	4112	['conv3d_15[0][0]']
concatenate_3 (Concatenate)	(None, 128, 128, 12,8,32)	0	['conv3d_transpose_3[0][0]', 'conv3d_1[0][0]']
conv3d_16 (Conv3D)	(None, 128, 128, 12, 8, 16)	13840	['concatenate_3[0][0]']
dropout_8 (Dropout)	(None, 128, 128, 12, 8, 16)	0	['conv3d_16[0][0]']
conv3d_17 (Conv3D)	(None, 128, 128, 12, 8, 16)	6928	['dropout_8[0][0]']
conv3d_18 (Conv3D)	(None, 128, 128, 12, 8, 4)	68	['conv3d_17[0][0]']

Chapter 5

5 Results and Discussion

5.1 System Configuration – Implementation details

To determine the appropriateness of the proposed model for the task of segmenting brain tumors based on MRI and fMRI scans, experiments were conducted. The experiment was conducted in two different machines/systems. Data pre-processing, generation of samples, as well as training of the model was done on a platform built with Intel(R) Core(TM) i5-10300H CPU at 2.50 GHz, 8 GB of memory, and an Nvidia 1650Ti Graphics board.

```
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 510.108.03   Driver Version: 510.108.03   CUDA Version: 11.6     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  NVIDIA GeForce ...  Off  | 00000000:01:00.0  On |                  N/A |
| N/A   51C    P8     5W /  N/A |     54MiB /  4096MiB |     32%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|    0   N/A  N/A      2211      G   /usr/lib/xorg/Xorg                 53MiB |
+-----------------------------------------------------------------------------+
```

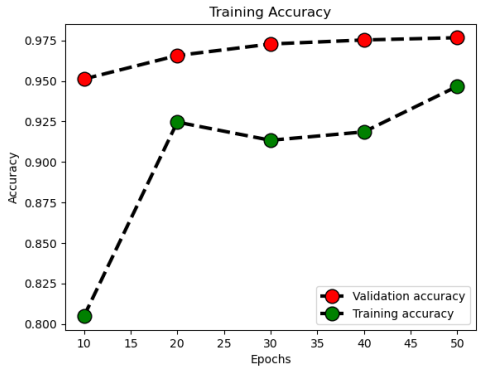
Figure 16: System Specifications w/ CUDA support

5.2 Assessment of the Model

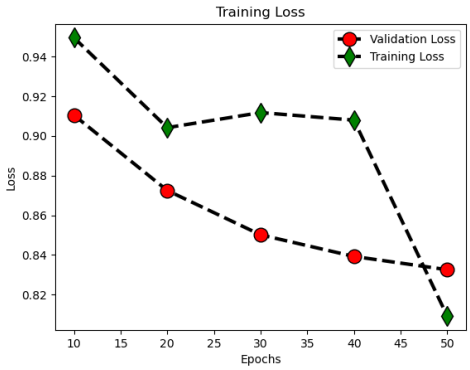
During the training phase, the u-net started to show some convergence after 35 Epochs. In the first 10-20 Epochs, there were a lot of randomnesses. The batch-wise training is represented in the following table:

Table 2: Performance statistics for every 10 Epochs

Epochs	Loss	Accuracy	Val. Loss	Val. Acc.	IoU Score	Val. IoU Score
10	91.05	95.12	94.96	80.48	33.98	28.56
20	87.25	96.57	90.42	92.46	46.06	37.84
30	85.01	97.28	91.18	91.34	52.89	36.81
40	83.92	97.53	90.80	91.86	56.78	37.46
50	83.25	97.67	80.90	94.67	59.48	42.43



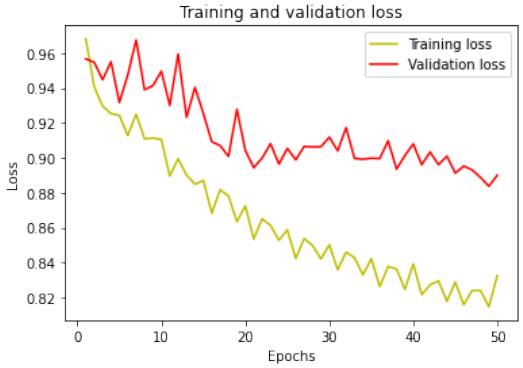
(a) Training and validation accuracy



(b) Training and validation loss

Figure 17: Performance statistics for every 10 epochs

(a) Training and validation accuracy



(b) Training and validation loss

Figure 18: Performance Evaluation

For the Training batches and over 50 Epochs with steps per epoch as 50, the model provides an Accuracy of 97.67% and a Training Loss of 83.25%, followed by a Validation Accuracy of 94.67% and a Validation Loss of 80.90%.

5.3 Metric for evaluating the Segmented Masks

It's not that easy to validate the performance of Semantic Segmentation on Bio-Medical Imaging, so for that, there is a major non-homogeneity while choosing the evaluation metric/parameter. For that reason, IoU (*Intersection over Union*) has been selected as the standard measure for evaluating the result of the ROI. Where at the end of Training, the Mean IoU is 42.82%. The

Equation for IoU and Mean IoU is given below:

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Intersection} \text{ and } Mean\ IoU = \frac{\sum_{i=1}^{n} No.\ of\ IoU}{Total\ no.\ of\ candidate\ for\ IoU}$$

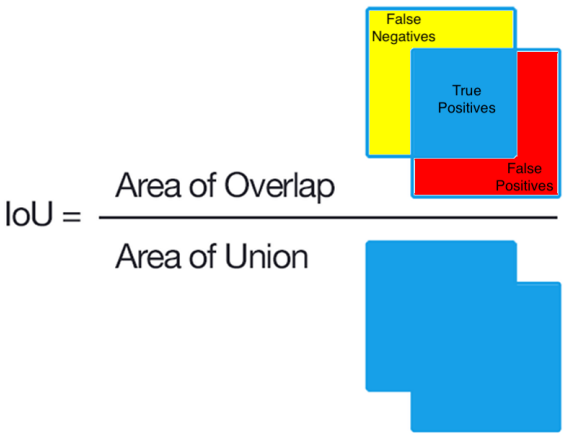


Figure 19: Diagram of IoU

Chapter 6

6 Conclusion

In this work, we presented a novel approach to semantically segment Brain Tumors from large-scale multi-band 3D volumes using Attention based U-Net architecture. Brain MR images provide very pertinent information as to whether there are any outlines concerning a brain tumor. This project aims to perform semantic segmentation of brain tumors on 3D MRI images using Attention based U-Net architecture. Each pixel in the image is assigned a semantic class, and the abnormal area is localized using the Attention based U-Net architecture. A segmented mask of the tumor is the output of our model. After the experiments, it has been found that the model performs well and yields a Train Accuracy of 97.67% and a Training Loss of 83.25%, followed by a Validation Accuracy of 94.67% and a Validation Loss of 80.90%. There are several interesting aspects of this project, but the most distinctive one is the use of 3D scans to produce more accurate results rather than conventional imagery, which is a novel approach. A patient's data can be used to determine the extent of the tumor and to take supervised precautions, such as the time of resection, the age at which the tumor first appeared, the stage of the tumor, etc.

Chapter 7

7 Future Work

We want to improve the segmentation of brain tumors performance of the attention-based U-Net. This could involve experimenting with various network structures, such as deeper or wider U-Net variations, or introducing additional attention mechanisms to capture more minute details. Additionally, we want to create methods for calculating the degree of uncertainty in the attention-based U-Net's forecasts. In medical applications, quantifying uncertainty is crucial for giving physicians confidence metrics for making decisions. For this, one can look into ensemble methods, dropout-based uncertainty estimation, or Bayesian approaches. To learn more about the model's decision-making process, we would also like to look into techniques for visualizing and interpreting the attention maps the network produces. This might promote better comprehension of the segmentation results and help develop trust with clinicians. Finally, using a variety of clinical datasets, we plan to thoroughly assess and validate the attention-based U-Net. To evaluate the model's efficacy and clinical utility, work with medical professionals. Additionally, look at ways to incorporate the segmentation model that has been created into clinical workflows, such as integrating it with current radiology systems or creating user-friendly interfaces for accurate and reliable tumor segmentation.

References

[1] Md Khairul Islam Et. al., “Brain tumor detection in MR image using superpixels, principal component analysis and template-based K-means clustering algorithm", Machine Learning with Applications.

[2] Anuja Arora Et. al., “Brain Tumor Segmentation of MRI Images Using Processed Image Driven U-Net Architecture", MDPI

[3] Aaswad Sawant Et. al., “Brain tumor detection and classification using machine learning: a comprehensive survey", Springer 8, pages3161–3183 (2022)

[4] Mrs. Madhuri Gurale Et. al., “Brain Tumor Detection using Image Processing, ML & NLP", International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056, p-ISSN: 2395-0072

[5] Naveen V Et.al., “Brain Tumor Detection Using Machine Learning Approach", International Research Journal of Modernization in Engineering Technology and Science, e-ISSN: 2582-5208

[6] Mrs Komal Sharma Et. al., “Brain Tumor Detection based on Machine Learning Algorithms", International Journal of Computer Applications (0975 – 8887)

[7] Mr.Ginni Garg1 Et. al., “Brain Tumor Detection and Classification based on Hybrid Ensemble Classifie", (Arvix)

[8] AMRUTA PRAMOD HEBLI Et.al., “BRAIN TUMOR DETECTION USING IMAGE PROCESSING: A SURVEY", International Journal of Industrial Electronics and Electrical Engineering, ISSN: 2347-6982, p-ISSN: 2395-0072

[9] Aaswad Sawant Et. .al., “BRAIN CANCER DETECTION FROM MRI: A MACHINE LEARNING APPROACH (TENSORFLOW)", International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056

[10] Ebrahim Mohammed Senan Et.al., “ Early Diagnosis of Brain Tumour MRI Images Using Hybrid Techniques between Deep and Machine Learning",Hindawi Computational and Mathematical Methods in Medicine,Volume 2022, Article ID 8330833, 17 pages,https://doi.org/10.1155/2022/8330833

[11] Ali Mohammad Alqudah Et. al., “Brain Tumor Classification Using Deep Learning Technique - A Comparison between Cropped, Uncropped, and Segmented Lesion Images with Different Sizes",International Journal of Advanced Trends in Computer Science and Engineering, ISSN 2278-3091.

[12] Wessam M. Salama et. al., “A novel framework for brain tumor detection based on convolutional variational generative models", Spring 81, pages16441–16454 (2022).

[13] Gopal S Tandel et. al, “A Review on a Deep Learning Perspective in Brain Cancer Classification", MDPI.

[14] Venkatesan Rajinikanth et. al, “A Customized VGG19 Network with Concatenation of Deep and Handcrafted Features for Brain Tumor Detection", MDPI.

[15] Hareem Kibriya et. al, “Multiclass Brain Tumor Classification Using Convolutional Neural Network and Support Vector Machine", IEEE.

[16] Priyanka Datta et. al, “Transfer Learning-Based Brain Tumor Detection Using MR Images", Computational and Experimental Methods in Mechanical Engineering.

[17] Kapil Kumar Gupta et. al., “Depth Analysis of Different Medical Image Segmentation Techniques for Brain Tumor Detection ", Advances in Bioinformatics, Multimedia, and Electronics Circuits and Signals pp 197–214.

[18] Omar Abdullah Murshed Farhan Alnaggar et. al, “Brain tumor detection from 3D MRI using Hyper-Layer Convolutional Neural Networks and Hyper-Heuristic Extreme Learning Machine", Concurrency and Computation: Practice and ExperienceVolume 34, Issue 24 e7215.

[19] Navid Ghassemi et. al., “Deep neural network with generative adversarial networks pre-training for brain tumor classification based on MR images", Biomedical Signal Processing and Control.

[20] Himar Fabelo et. al., “SVM Optimization for Brain Tumor Identification Using Infrared Spectroscopic Samples", MDPI.

[21] Mehrdad Noori et. al., “Attention-Guided Version of 2D UNet for Automatic Brain Tumor Segmentation", (arXiv).

[22] Navchetan Awasthi et. al., “Multi-Thresold Attention U-Net(MTAU) based Model for Multimodal Brain Tumor Segmentation in MRI scans", (arXiv).

[23] Xiuling Gan et. al., “GAU-Net:U-Net Based on Global Attention Mechanism for brain tumor segmentation", Journal of Physics: Conference Series, pages 1861-012041(2021).

[24] Jie Hu et. al, “Squeeze-and-Excitation Networks", arXiv:1709.01507 [cs.CV].

[25] Olaf Ronneberger et. al, “U-Net: Convolutional Networks for Biomedical Image Segmentation", arXiv:1505.04597 [cs.CV].