
A Project Report on
Medical Image Segmentation for Brain Tumor
Detection Using U-Net

A Dissertation submitted to JNTU Hyderabad in partial
fulfillment of the academic requirements for the award of the
degree.

Bachelor of Technology
In
Computer Science and Engineering

Submitted by

VUYYURU NAMITHA

(20H51A0580)

AKSHAY TONDE

(20H51A05K0)

JARPULA SIVAKUMAR

(20H51A05N8)

Under the esteemed guidance

Mr. A. Vivekanand

(Associate Professor)



Department of Computer Science and Engineering
CMR COLLEGE OF ENGINEERING & TECHNOLOGY

(UGC Autonomous)

*Approved by AICTE *Affiliated to JNTUH *NAAC Accredited with A⁺ Grade

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD - 501401.

2020- 2024

CMR COLLEGE OF ENGINEERING & TECHNOLOGY

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD – 501401

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



CERTIFICATE

This is to certify that the Major Project report entitled entitled "**"Medical Image Segmentation for Brain Tumor Detection using U-Net"** being submitted by **V. Namitha (20H51A0580)**, **K. Akshay Tonde (20H51A05K0)**, **J. SivaKumar (20H51A05N8)** in partial fulfillment for the award of **Bachelor of Technology in COMPUTER SCIENCE AND ENGINEERING** is a record of bonafide work carried out under my guidance and supervision.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree.

Mr. A. Vivekanand
Associate Professor
Dept. of CSE

Dr. Siva Skandha Sanagala
Associate Professor and HOD
Dept. of CSE

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

With great pleasure, we want to take this opportunity to express our heartfelt gratitude to all the people who helped in making this project a grand success.

We are grateful to our guide **Mr. A. Vivekanand (Associate Professor)**, Department of Computer Science and Engineering for his valuable technical suggestions and guidance during the execution of this project work.

We would like to thank, **Dr. Siva Skandha Sanagala**, Head of the Department of Computer Science and Engineering, CMR College of Engineering and Technology, who is the major driving force to complete our project work successfully.

We are very grateful to **Dr. Ghanta Devadasu**, Dean-Academics, CMR College of Engineering and Technology, for his constant support and motivation in carrying out the project work successfully.

We are highly indebted to **Major Dr. V A Narayana**, Principal, CMR College of Engineering and Technology, for giving permission to carry out this project in a successful and fruitful way.

We would like to thank the **Teaching & Non- teaching** staff of Department of Computer Science and Engineering for their co-operation.

We express our sincere thanks to **Shri. Ch. Gopal Reddy**, Secretary& Correspondent, CMR Group of Institutions, and **Shri. Ch. Abhinav Reddy**, CEO, CMR Group of Institutions for their continuous care and support.

Finally, we extend thanks to our parents who stood behind us at different stages of this Project. We sincerely acknowledge and thank all those who gave support directly or indirectly in completion of this project work.

V. NAMITHA - 20H51A0580

T. AKSHAY - 20H51A05K0

J. SIVAKUMAR - 20H51A05N8

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	LIST OF FIGURES	iii
	LIST OF TABLES	iv
	ABSTRACT	v
1	INTRODUCTION	1-6
	1.1 Introduction	2
	1.2 Problem Statement	4
	1.3 Research Objective	4
	1.4 Project Scope and Limitations	5
2	BACKGROUND WORK	7-11
	2.1. Manual Segmentation	8
	2.1.1. Introduction	8
	2.1.2. Merits, Demerits and Challenges	8
	2.1.3. Implementation	8
	2.2. Thresholding and Region Growing	9
	2.2.1. Introduction	9
	2.2.2. Merits, Demerits and Challenges	9
	2.2.3. Implementation	9
	2.3. Region-Based Active Contour Models	10
	2.3.1. Introduction	10
	2.3.2. Merits, Demerits and Challenges	10
	2.3.3. Implementation	10
	2.4. Comparison Table	11
3	PROPOSED SYSTEM	12-30
	3.1. U-net's Algorithm	13
	3.2. Proposed System	15
	3.3. System Design	16
	3.4. Project Implementation	24
4	RESULTS & DISCUSSIONS	31-32
	4.1. Results Obtained	32

5	CONCLUSION	33-34
	5.1. Conclusion	33
6	REFERENCES	35-37
	6.1. References	35
7	APPENDIX	38-44
	7.1. Project Code	38
8	GITHUB LINK	45
9	DOI	45
10	PUBLISHED PAPER	46-55
11	CERTIFICATES	56-59

List of Figures

FIGURE NO.	TITLE	PAGE NO.
3.1.1	U-net's Algorithm	13
3.3.1	System Architecture	16
3.3.2	Class Diagram	17
3.3.3	Use case Diagram	18
3.3.4	Sequence Diagram	19
3.3.5	Component Diagram	20
3.3.6	Deployment Diagram	21
3.3.7	Activity Diagram	22
3.3.8	Dataflow Diagram	23
3.4.1	Dataset Folder	24
3.4.2	Contents of Folder	24
3.4.3	Sample Dataset	25
3.4.4	Contents of Sample	25
3.4.5	User Interface	26
3.4.6	Uploading Dataset	26
3.4.7	Dataset Uploaded	27
3.4.8	Generating Model	27
3.4.9	Output Shape Params	28
3.4.10	Uploading Sample Dataset	28
3.4.11	Segmentation of MRI Scans	29
3.4.12	Uploading sample Dataset	30
3.4.13	Segmentation of sample dataset	30
4.1.1	Dice Score Graph	32

ABSTRACT

Brain tumors encompass a wide spectrum of growths that can occur within the cranial cavity. These tumors may be benign or malignant, and their early detection is vital for initiating timely medical interventions. MRI, a non-invasive imaging modality, provides detailed anatomical information, making it an invaluable tool for diagnosing and monitoring brain tumors. However, the accurate identification of tumor boundaries from complex MRI data remains a complex and demanding task. The project utilizes a sophisticated image segmentation method to automatically identify and outline tumor regions in MRI scans. The technique is based on the principles of deep learning, a subset of artificial intelligence, which enables computers to learn and recognize patterns from large datasets. This learning process enables the algorithm to distinguish between healthy brain tissue and abnormal tumor growths, aiding medical professionals in making informed decisions. The U-Net architecture's unique design, characterized by a U-shaped structure with contracting and expansive paths, enables it to effectively capture intricate structures and fine-grained details in medical images. Leveraging a diverse dataset of multi-modal brain MRI scans, the U-Net algorithm is trained to automatically segment tumor regions, providing pixel-level precision.

CHAPTER 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1. INTRODUCTION

Medical image segmentation is a crucial task in modern healthcare, facilitating the precise delineation of anatomical structures from various imaging modalities. Magnetic Resonance Imaging (MRI), known for its high-resolution visualization, plays a pivotal role in clinical diagnostics. However, manual segmentation of MRI images is time-consuming and subjective, highlighting the need for automated approaches. One notable algorithm in this domain is the U-Net architecture, introduced by Ronneberger et al. in their seminal work "U-Net: Convolutional Networks for Biomedical Image Segmentation" (Ronneberger, 2015) [1]. The U-Net's unique encoder-decoder design is adept at accurately localizing and segmenting intricate anatomical structures, making it highly suitable for medical image analysis. The U-Net architecture utilizes an encoder to extract contextual information and abstract features, while the decoder refines localization through up-sampling and feature concatenation. This structure enables U-Net to deliver precise segmentations, even in complex anatomical regions. Automated segmentation has shown significant promise in transforming medical image analysis, with U-Net emerging as a powerful tool in this field. The adoption of deep learning techniques in medical imaging has also received considerable attention, further emphasizing the importance of this project (Litjens et al., 2017 [2]; Greenspan et al., 2016 [3]). Accurate segmentation of tumors through medical images is particularly important as it provides essential information for the analysis and diagnosis of cancer, as well as for planning treatment options and monitoring disease progression. Brain tumors, among the deadliest cancers worldwide, are categorized into primary and secondary types based on their origin. The most common histological type of primary brain cancer is glioma, which originates from the brain's glial cells and accounts for 80% of all malignant brain tumors. Gliomas can be slow-progressing low-grade (LGG), which have a better prognosis, or the more aggressive and infiltrative high-grade gliomas (HGG) or glioblastomas, requiring immediate treatment. These tumors are associated with significant morbidity, with the median survival for a patient with glioblastoma being only about 14 months and a 5-year survival rate near zero despite maximal surgical and medical therapy. Thus, timely diagnosis is crucial for effective treatment. MRI is a preferred technique widely used by radiologists for evaluating and assessing brain tumors. Wu, B., et al. (2024) [4] It provides several complementary 3D MRI modalities based on different degrees of excitation and repetition times, i.e., T1-weighted, post-contrast T1-weighted (T1ce), T2-weighted, and Fluid-Attenuated Inversion Recovery (FLAIR). The highlighted subregions of the

tumor across different intensities of these sequences, such as the whole tumor (inclusive of infiltrative edema), are more prominent in FLAIR and T2 modalities, while T1 and T1ce images show the tumor core exclusive of peritumoral edema. This allows for the combined use of these scans and the complementary information they provide for detecting different tumor subregions. The Multimodal Brain Tumor Segmentation Challenge (BraTS) offers a platform to evaluate the development of machine learning models for tumor segmentation by providing participants with extensive dataset of 3D MRI images of gliomas (both LGG and HGG) and associated ground truths annotated by expert physicians. Nguyen, P. X., et al. (2019) [5] The multimodal scans are used for training and validating neural networks designed for this specific segmentation task. Manual delineation of brain tumor sub regions from MRI scans is subjective, time-consuming, and prone to variability. Therefore, automated segmentation of gliomas from multimodal MRI images can assist physicians in speeding up diagnosis and surgical planning, providing an accurate, reproducible solution for further tumor analysis and monitoring. Classical methods of automated brain tumor segmentation rely on feature engineering, which involves extracting handcrafted features from input images followed by classifier training. In contrast, unsupervised learning algorithms and deep learning models excel by automatically learning a hierarchy of feature representation, with Convolutional Neural Networks (CNNs) regarded as the state-of-the-art methods for brain tumor image segmentation. They automatically learn the most useful and relevant features. However, accurate segmentation of tumors remains challenging due to the heterogeneity in shape, size, and appearance of gliomas, as well as the ambiguous and fuzzy boundaries between cancer and brain tissue. Huang, C., et al. (2019) [6] The intensity variability of MRI data further complicates this issue. Therefore, there is room for improvement and further exploration for better segmentation techniques and accuracy. In this work, we utilize multiple 3D CNN models for brain tumor segmentation from multimodal MRI scans and ensemble their probability maps for more stable predictions. Lu, H., et al. (2022) [7] The networks are trained separately, with hyper parameters optimized for each model, on the training dataset acquired from the 2019 Brain Tumor Segmentation (BraTS) challenge. A rigorous evaluation on the BraTS validation set resulted in the proposed ensemble achieving Dice scores of 0.750, 0.906, and 0.846 for enhancing tumor, whole tumor, and tumor core, respectively.

1.2 PROBLEM STATEMENT

The accurate identification and delineation of tumor boundaries in complex MRI data remains a challenging task. Early detection of brain tumors is crucial for timely medical interventions. While MRI is a valuable non-invasive imaging modality, manual segmentation of tumor regions is time-consuming and prone to human error. Therefore, there is a pressing need for an automated and accurate method to identify and outline tumor regions in MRI scans, facilitating more efficient diagnosis and monitoring of brain tumors. This project aims to address this challenge by employing a sophisticated image segmentation technique based on deep learning principles, specifically utilizing the U-Net architecture, to achieve pixel-level precision in segmenting tumor regions.

1.3 RESEARCH OBJECTIVE

Automated segmentation of brain tumors from multimodal MR images is crucial for analyzing and monitoring disease progression. Gliomas, known for their malignancy and heterogeneity, necessitates efficient and precise segmentation techniques for the successful delineation of tumors into intra-tumoral classes. Deep learning algorithms excel in tasks of semantic segmentation, surpassing more traditional, context-based computer vision approaches. Widely employed for biomedical image segmentation, Convolutional Neural Networks (CNNs) have notably enhanced the accuracy of brain tumor segmentation. In this paper, we introduce an ensemble of two segmentation networks: a 3D CNN and a U-Net, employing a simple yet effective combinative approach that yields improved and more precise predictions. Both models were trained independently on the BraTS-19 challenge dataset and evaluated, producing segmentation maps that significantly varied from each other in terms of segmented tumor sub-regions. These maps were then combined in a novel manner to achieve the final prediction. Our proposed ensemble achieved Dice scores of 0.750, 0.906, and 0.846 for enhancing tumor, whole tumor, and tumor core, respectively, on the validation set. This performance is favorable in comparison to the current state-of-the-art architectures available.

1.3 PROJECT SCOPE

1. Objective: The project aims to develop an automatic brain tumour segmentation system using the U-Net deep learning architecture applied to multi-modal MRI scans. This system will assist in the accurate identification and outlining of tumour regions within the cranial cavity.
2. Medical Imaging Focus: The project focuses on the use of MRI scans as the primary source of medical imaging data. It intends to enhance the precision of brain tumour segmentation within these scans.
3. Deep Learning and U-Net: The project utilizes deep learning techniques, particularly the U-Net architecture, to automatically detect and outline tumour regions. U-Net's unique design is leveraged for its ability to capture intricate structures and fine-grained details in medical images.
4. Dataset Utilization: The project involves the use of a diverse dataset of multi-modal brain MRI scans for training the U-Net algorithm. The dataset is essential for enabling the algorithm to distinguish between healthy brain tissue and abnormal tumour growths effectively.

Limitations:

1. Data Availability: The project's success heavily relies on the availability of high-quality and diverse MRI datasets. The accuracy of the segmentation model is limited by the quality and representativeness of the training data.
2. Algorithm Generalization: The deep learning model, while powerful, may not generalize well to all possible variations in tumour appearances. Its effectiveness in segmenting different types and sizes of tumours is subject to the quality and diversity of the training data.
3. Performance Dependency: The performance of the system is dependent on the quality and resolution of the MRI scans. Lower-quality scans may yield less accurate segmentation results.
4. Algorithm Computation: The computational resources required for training and running deep learning models can be substantial. The project may be limited by the availability of such resources.
5. Human Oversight: Although the project aims to automate the tumour segmentation process, it is important to note that the results produced by the algorithm may still require

human oversight and validation, particularly in a clinical setting.

6. Legal and Ethical Considerations: The project may be subject to legal and ethical considerations, including patient consent, data privacy, and regulatory compliance when working with medical imaging data. These considerations may influence the implementation and deployment of the system.
7. Clinical Integration: The integration of the developed system into clinical practice may have limitations related to compatibility with existing healthcare infrastructure and the willingness of medical professionals to adopt and trust the technology.
8. Ongoing Research: The project may not cover all possible nuances and complexities of brain tumour segmentation, and it is subject to ongoing research and advancements in the field of medical imaging and deep learning.

CHAPTER 2

BACKGROUND

WORK

CHAPTER 2

BACKGROUND WORK

2.1 Manual Segmentation:

2.1.1. Introduction

In this section we have studied various implementation of Medical image segmentation using U-net's algorithm and we have summarized our findings that we concluded by researching and referencing various papers. They are as below.

2.1.2 Merits, Demerits, and Challenges

Merits:

- **High Accuracy:** Manual segmentation is Arora, A., et al. (2021) [8] capable of achieving high levels of accuracy,especially when tumors are well-defined in the scans.
- **Clinical Expertise:** Radiologists can make clinical judgments during the segmentationprocess, taking into account a patient's medical history and context.

Demerits:

- **Subjective Variability:** The accuracy can vary between different experts, leading to inter-and Yin, X. X., et al. (2022) [9] intra-observer variability.
- **Time-Consuming:** Manual segmentation is a time-consuming process, which may not besuitable for situations requiring rapid decision-making.

Challenges:

- **Variability:** Variability in segmentations can lead to differences in treatment decisions.Resource-Intensive: It requires skilled human Sun, J., et al. (2020) [10] resources and is often impractical for large-scale applications.

2.1.3 Implementation

Radiologists use specialized medical imaging software to manually draw and outline tumor boundaries. The process is labor-intensive and typically integrated into the clinical workflow.

2.2 Thresholding and Region Growing:

2.2.1 Introduction

Thresholding and region growing are traditional computer-assisted methods for tumor segmentation. They rely on intensity thresholds and spatial information to classify and segment tumor regions in MRI scans.

2.2.2 Merits, Demerits and Challenges

Merits:

Simplicity: These methods are Jwaid, W. M., et al [11] relatively straight forward to implement and do not require extensive computational resources.

Speed: Depending on the specific technique, they can provide relatively fast results.

Demerits:

Complex Cases: Nguyen, P. X., et al [12] These methods may struggle with complex tumor shapes, heterogeneous appearances, or small lesions.

Noise Sensitivity: MRI scans can contain noise, leading to inaccurate segmentations with these methods.

Challenges:

Accuracy: Achieving high accuracy, especially for intricate or irregular tumor shapes, can be challenging.

Robustness: The methods may not adapt well to variations in MRI scan quality or pathology.

2.2.3 Implementation

Thresholding methods use intensity thresholds to classify pixels as tumor or non-tumor. Region growing algorithms start from seed points and expand regions based on pixel similarity. These techniques are typically implemented using image processing libraries and tools.

2.3 Region-Based Active Contour Models:

2.3.1 Introduction

Region-based active contour models, also known as snakes, are segmentation methods that use deformable models to iteratively adjust contours to segment objects of interest, such as brain tumors.

2.3.2 Merits, Demerits and Challenges

Merits:

- **Adaptability:** Zheng, P., et al. (2022) [13] These models can adapt to varying tumor shapes and sizes.
- **Object Interaction:** They can handle multiple objects in the image and segment them individually.

Demerits:

- **Initialization Sensitivity:** The accuracy of the segmentation can be sensitive to the initial contour placement.
- **Computational Load:** These Sun, J., et al. [14] models can be computationally intensive, particularly for real-time applications.

Challenges:

- **Initialization:** Proper initialization of the contour is crucial for accurate results.
- **Computational Resources:** Implementing these models can require significant computational resources.

2.3.3 Implementation

Active contour models use energy minimization to iteratively adjust contours. The initialization of the contour is a critical step in the segmentation process. These models can be implemented using various image analysis and computer vision libraries.

2.4 Comparison Table

S.No	Authors and Journal Name & Year of publication	Problem Statement	Name of the Proposed solution/Method	Solution	Remarks
1	Muhammad Zawish; Asad Ali Siyal; Kainat Ahmed; Aiman Khalil; Sheeraz Memon 2018 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube)	Biomedical imaging has advanced, yet tumor segmentation remains a challenge due to tissue overlap. Accurate segmentation is vital for tumor treatment.	Brain Tumor Segmentation in MRI images using Chan Vese Technique in MATLAB	Novel tumor segmentation method using total variation and Chan Vese contours for precise brain MRI tumor delineation, benefiting healthcare and research.	Tumor segmentation is a critical issue in medical imaging. The Chan Vese active contour method, with its dual forces, effectively addresses this challenge. Extensive testing confirms its applicability to various medical imaging modalities, promising broader biomedical research applications.
2	Kapil Kumar Gupta; Namrata Dhanda; Upendra Kumar 2018 4th International Conference on Computing Communication and Automation (ICCCA)	Accurate CT and MR image segmentation for tumor detection is crucial but challenging in medical imaging.	A Comparative Study of Medical Image Segmentation Techniques for Brain Tumor Detection	Researchers review and compare segmentation techniques, considering factors like segmentation time, accuracy, and sensitivity.	Accurate segmentation enhances medical diagnosis and treatment through non-invasive techniques like Ultrasound, MRI, and CT-Scan.
3	Chuanlu Lin; Yi Wang; Tianfu Wang; Dong Ni 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)	Conventional LSD methods distort pathological regions during MR brain image recovery and segmentation.	Segmentation and Recovery of Pathological MR Brain Images Using Transformed Low-Rank and Structured Sparse Decomposition	The TLS2D method provides robust extraction of pathological regions with satisfactory image recovery and segmentation performance.	TLS2D enhances MR image analysis, particularly in brain tumor detection, offering more accurate and reliable results.
4	Hayder Saad Abdulbaqi; Mohd Zubir Mat; Ahmad Fairuz Omar; Iskandar Shahrim Bin Mustafa; Loay Kadom Abood 2014 IEEE Student Conference on Research and Development	Manual MRI brain tumor segmentation is time-consuming; accurate detection is vital. Fast and accurate tumor detection is challenging.	Detecting brain tumor in Magnetic Resonance Images using Hidden Markov Random Fields and Threshold techniques	The proposed hybrid method combines HMRF and Threshold techniques, improving brain tumor detection on MRI images.	This approach enhances segmentation and preserves tissue boundaries, promising better diagnosis and treatment.
5	M. Kadkhodaei; S. Samavi; N. Karimi; H. Mohaghegh; S. M.R. Soroushmehr; K. Ward; A. All; K. Najarian 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)	Challenges in brain tumor segmentation persist. Glioma tumors vary; automatic segmentation is crucial. Proposed method enhances segmentation, improving accuracy.	Automatic segmentation of multimodal brain tumor images based on classification of supervoxels	Proposed method enhances glioma tumor segmentation using preprocessing, saliency-based feature extraction, and texture features, outperforming existing algorithms.	Automation aids robust brain tumor diagnosis. Enhanced feature extraction and boundary alignment enhance performance over a comparable algorithm.

(Table 2.4.1 Comparison Table of various research papers)

CHAPTER 3

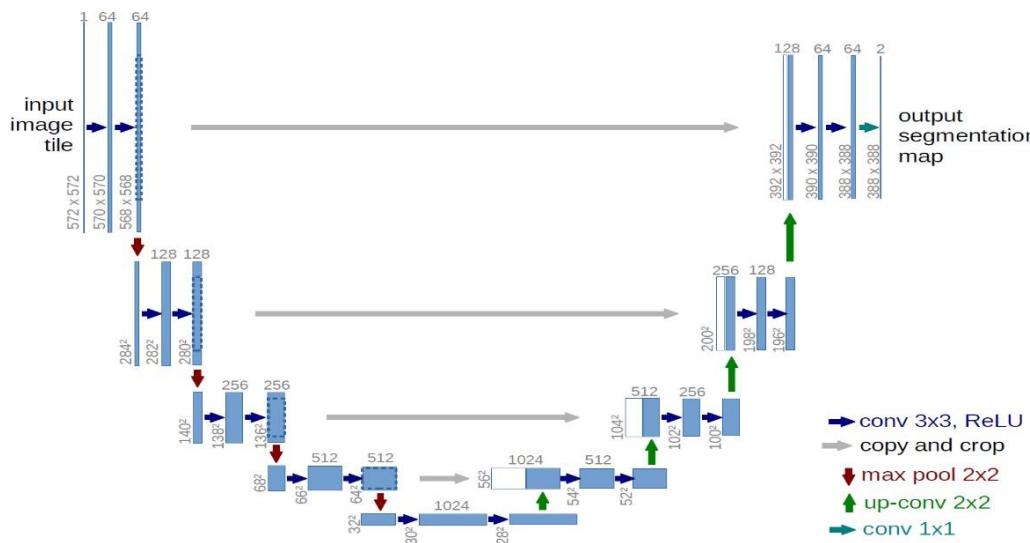
PROPOSED SYSTEM

CHAPTER 3

PROPOSED SYSTEM

3.1 U-net's Algorithm:

U-Net is a convolutional neural network (CNN) architecture that was designed for biomedical image segmentation, including tasks such as brain tumor segmentation. Originally introduced by Olaf Ronneberger, Philipp Fischer, and Thomas Brox in their paper "U-Net: Convolutional Networks for Biomedical Image Segmentation," U-Net has become a popular choice due to its efficiency and effectiveness in handling a variety of segmentation tasks, especially when dealing with a limited amount of data.



(Fig:3.1.1 U-net's Architecture)

Key Features of U-Net:

Symmetrical Structure: The architecture resembles the letter "U," which is the origin of its name. This structure comprises a contracting path to capture context and a symmetric expanding path that enables precise localization.

Contracting Path: The left side of the U, which is responsible for capturing the context in the image. It consists of a series of convolutional and max pooling layers that reduce the spatial dimensions of the input image while increasing the feature maps.

Expanding Path: The right side of the U, which is responsible for precise localization, using transposed convolutions to increase the resolution of the output. This path also includes concatenation with corresponding feature maps from the contracting path to ensure that the network can use features from

both local and global context.

Skip Connections: These are crucial for the U-Net architecture. They connect feature maps from the contracting path to the expanding path, helping the network to recover the spatial information lost during down sampling.

Fewer Training Samples: U-Net is designed to work well even with a very small number of training images, leveraging extensive data augmentation techniques to enhance the training set.

Structure of U-Net:

The structure of U-Net can be divided into two main parts: the contracting/down sampling path and the expanding/up sampling path.

Contracting Path: It typically starts with an input image passed through repeated blocks of two 3×3 convolutions followed by a rectified linear unit (ReLU) activation function, and then a 2×2 max pooling operation with stride 2 for down sampling. At each down sampling step, the number of feature channels is doubled.

Bottleneck: This is the bridge between the contracting and expanding paths. It usually consists of two 3×3 convolutions, each followed by a ReLU.

Expanding Path: Each step in the expanding path consists of an up sampling of the feature map followed by a 2×2 convolution (“up-convolution”) that halves the number of feature channels, a concatenation with the correspondingly cropped feature map from the contracting path, and two 3×3 convolutions, each followed by a ReLU. The cropping is necessary due to the loss of border pixels in every convolution.

Output Layer: Finally, a 1×1 convolution is used to map each 64-component feature vector to the desired number of classes. In the case of binary segmentation, such as distinguishing the tumor from the non-tumor region, the output layer would have a single channel.

3.2 Proposed System:

To implement this project we are using 4 different images these images are called FLAIR, T1, T2, and T1CE, and the label segmented image. The multi-institutional dataset, acquired from 19 different contributors, contains multimodal MRI scans of each patient, namely T1, T1 contrast-enhanced (T1ce), T2-weighted (T2), and Fluid Attenuated Inversion Recovery (FLAIR), from which the tumoural sub-regions are segmented. The data is processed to overcome discrepancies such that they are skull- stripped.

Advantage:

- 1) More Accuracy.

Modules:

- 1) Upload BRATS Dataset
- 2) Generate CNN & UNET Model
- 3) Upload Test Image & Segmentation
- 4) Dice Similarity Graph

1. Upload BRATS Dataset:

Upload BRATS Dataset is the first module of our project, it is used to upload the BRATS dataset.

2. Generate CNN & UNET Model:

Generate CNN & UNET Model is the second module of our project, it is used to models and console to see CNN and UNET layer details. we can see models are using different size images to filter them and to get the best features from it to build an efficient model and now the model is generated.

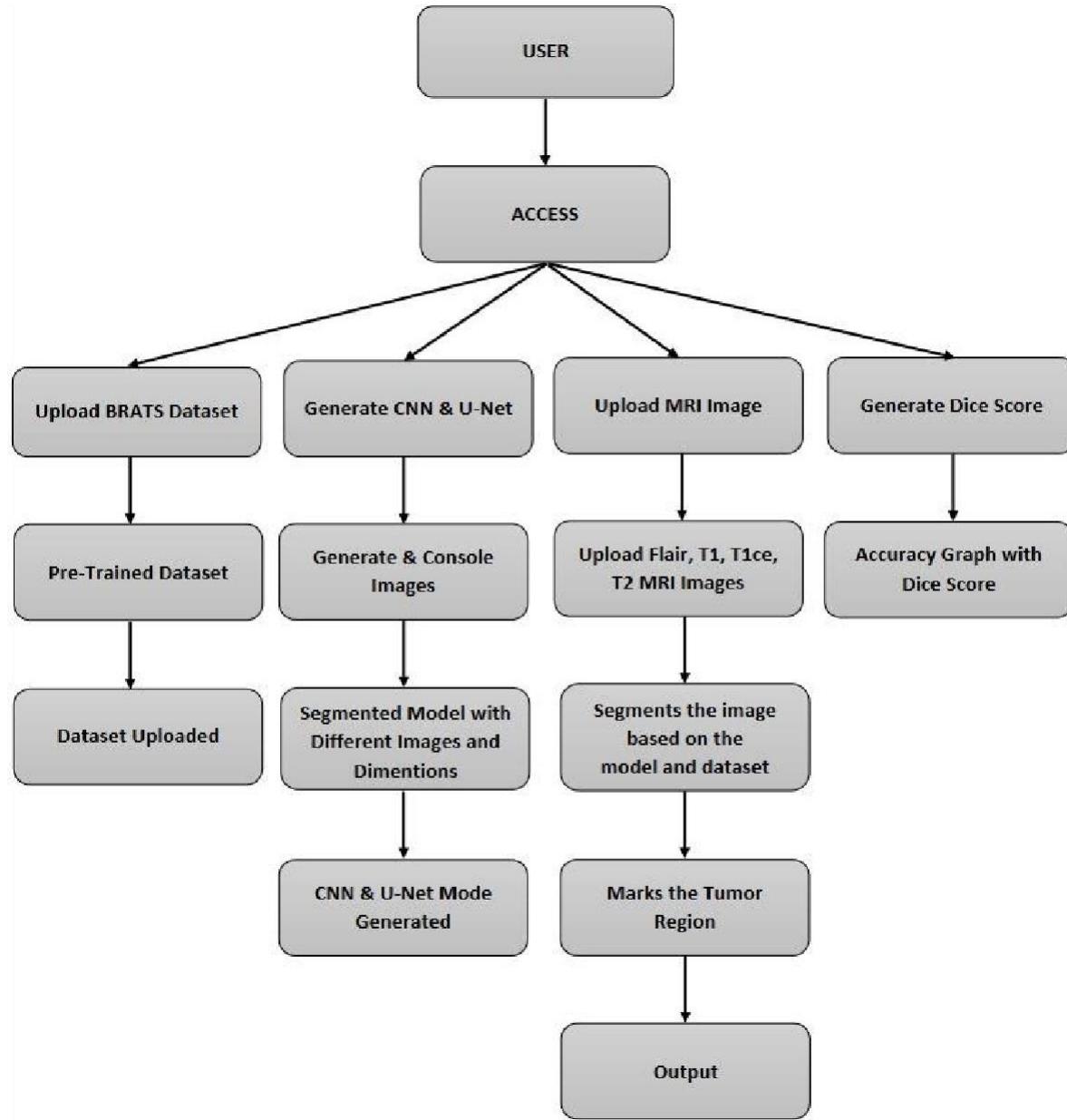
3. Upload Test Image & Segmentation:

Upload Test Image & Segmentation is the third module of our project and then upload test samples to get segmented output. selecting and uploading the ‘Sample1’ folder and then clicking on ‘Select Folder’button to get the below output top 4 images are the input images such as FLAIR, T1, T2, and T1CE and 5th image is the predicted image with the segmented part showing in red color and this algorithm correctly detecting and marking tumor area and now test with another image.

4. Dice Similarity Graph:

To build the CNN and UNET model we took 50 epochs or iterations and at each iteration DICE score between training and testing images got better and better and we got final dice score of $0.8 * 100 = 80\%$. In above graph x-axis represents epoch and y-axis represents dice score.

3.3 System Design

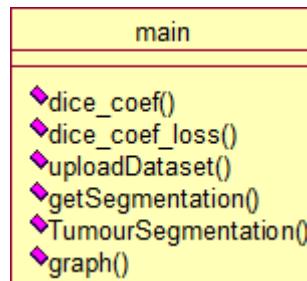


(Fig:3.3.1 System Architecture)

Class Diagram:

The class diagram is the main building block of object-oriented modeling. It is used both for general conceptual modeling of the systematics of the application and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed. In the diagram, classes are represented with boxes which contain three parts:

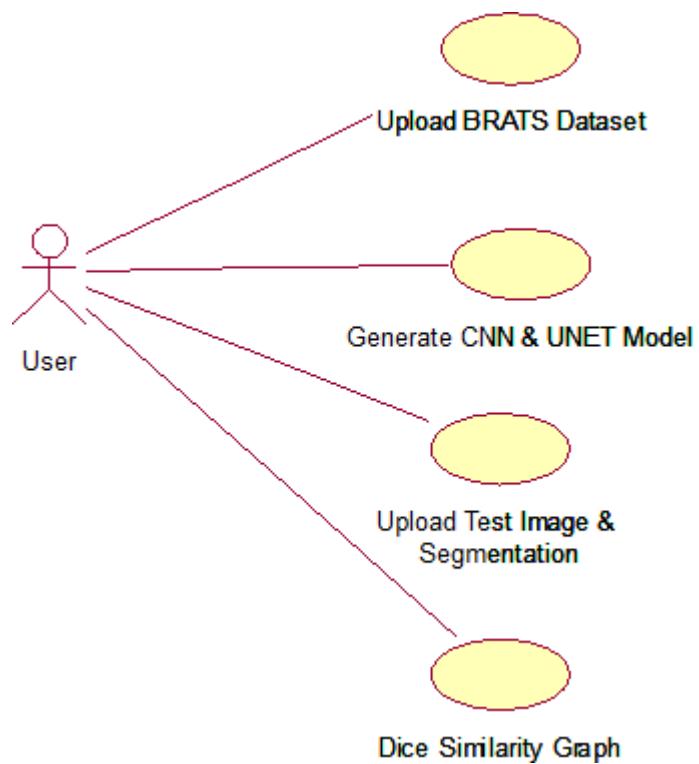
- The upper part holds the name of the class
- The middle part contains the attributes of the class
- The bottom part gives the methods or operations the class can take or undertake.



(Fig: 3.3.2 Class Diagram)

Use case Diagram:

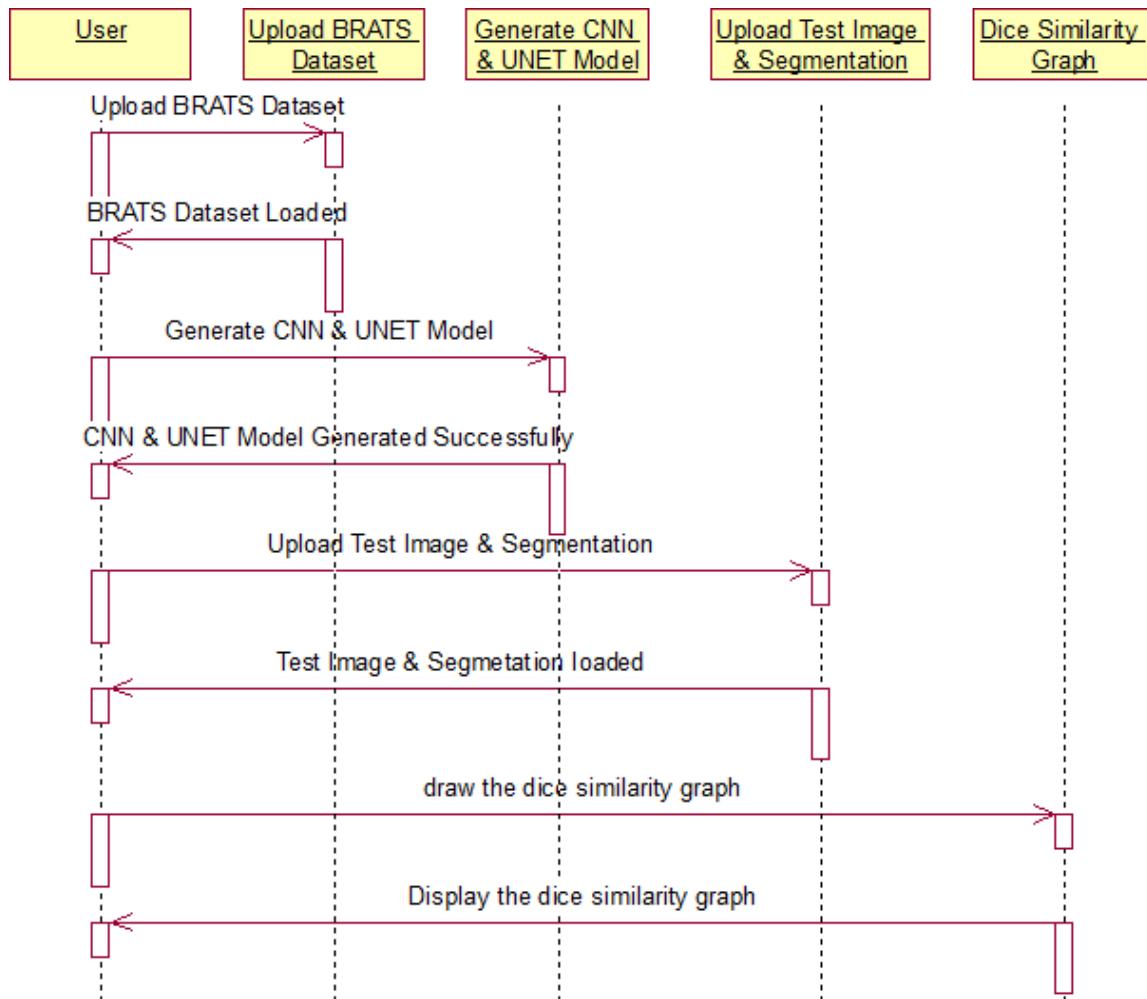
A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. This type of diagram is typically used in conjunction with the textual use case and will often be accompanied by other types of diagrams as well.



(Fig: 3.3.3 Use Case Diagram)

Sequence Diagram:

A sequence diagram is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

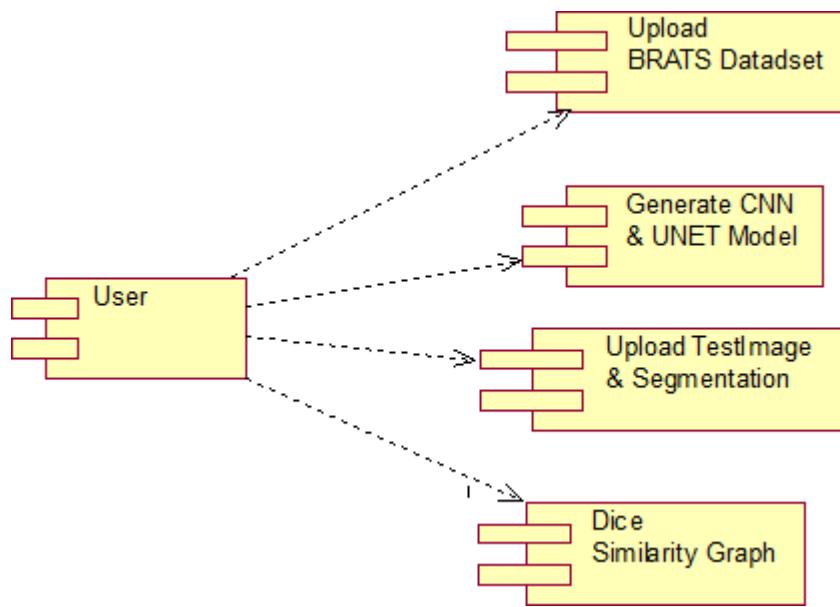


(Fig: 3.3.4 Sequence Diagram)

Component Diagram:

In the Unified Modeling Language, a component diagram depicts how components are wired together to form larger components and or software systems. They are used to illustrate the structure of arbitrarily complex systems.

Components are wired together by using an assembly connector to connect the required interface of one component with the provided interface of another component. This illustrates the service consumer - service provider relationship between the two components.

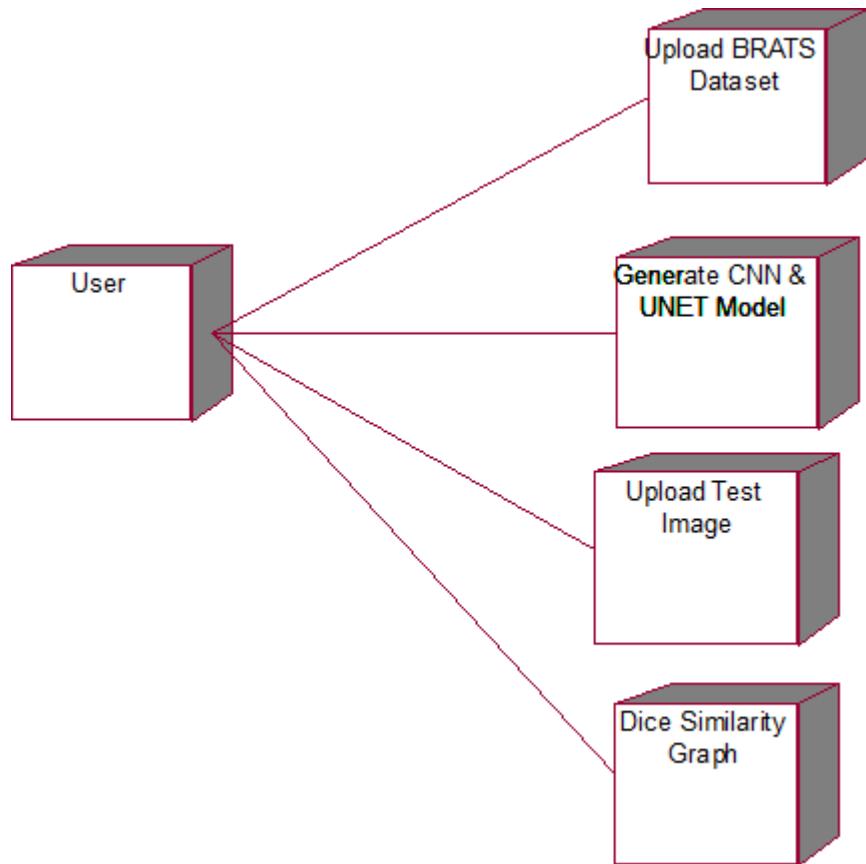


(Fig: 3.3.5 Component Diagram)

Deployment Diagram:

A deployment diagram in the Unified Modeling Language models the *physical* deployment of artifacts on nodes. To describe a website, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI).

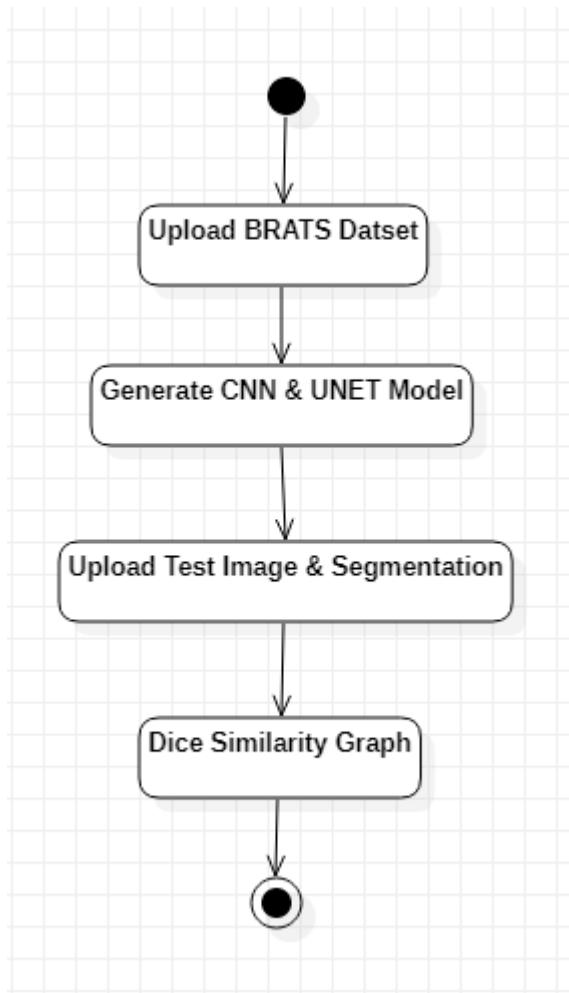
The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have sub-nodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.



(Fig: 3.3.6 Deployment Diagram)

Activity Diagram:

The activity diagram is another important diagram in UML to describe the dynamic aspects of the system. It is a flow chart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. So, the control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent.



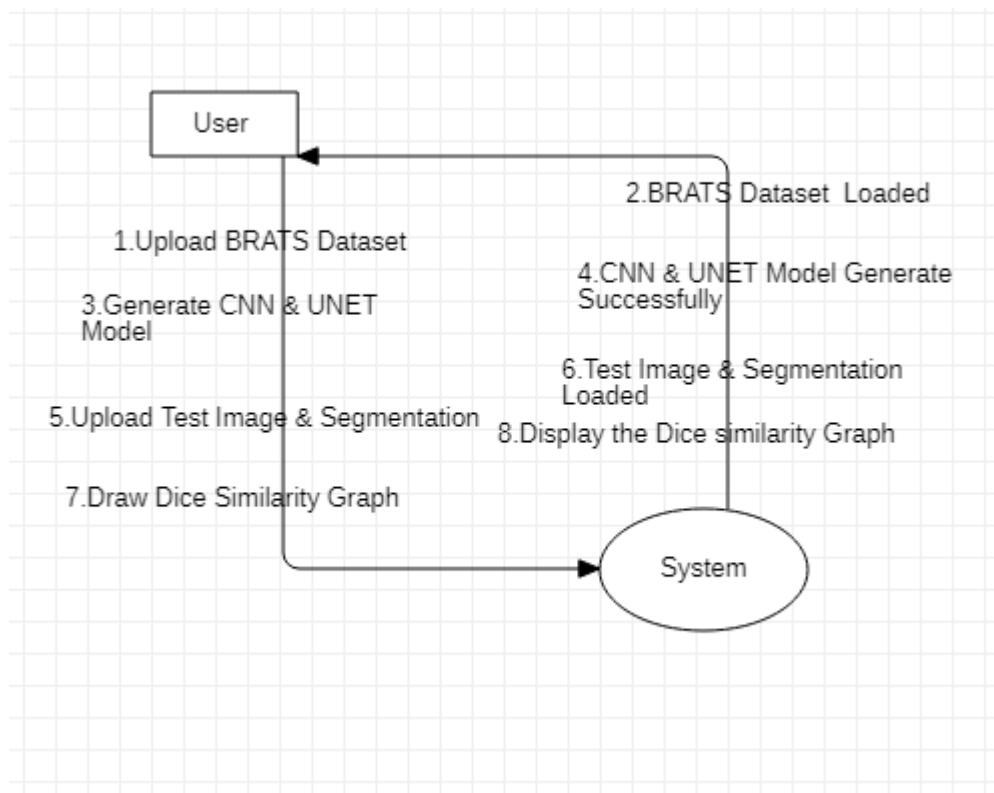
(Fig: 3.3.7 Activity Diagram)

Data Flow Diagram:

Data flow diagrams illustrate how data is processed by a system in terms of inputs and outputs. Data flow diagrams can be used to provide a clear representation of any business function. The technique starts with an overall picture of the business and continues by analyzing each of the functional areas of interest. This analysis can be carried out in precisely the level of detail required. The technique exploits a method called top-down expansion to conduct the analysis in a targeted way. As the name suggests, Data Flow Diagram (DFD) is an illustration that explicates the passage of information in a process. A DFD can be easily drawn using simple symbols.

Additionally, complicated processes can be easily automated by creating DFDs using easy-to-use, free downloadable diagramming tools. A DFD is a model for constructing and analyzing information processes. DFD illustrates the flow of information in a process depending upon the inputs and outputs. A DFD can also be referred to as a Process Model. A DFD demonstrates a business or technical process with

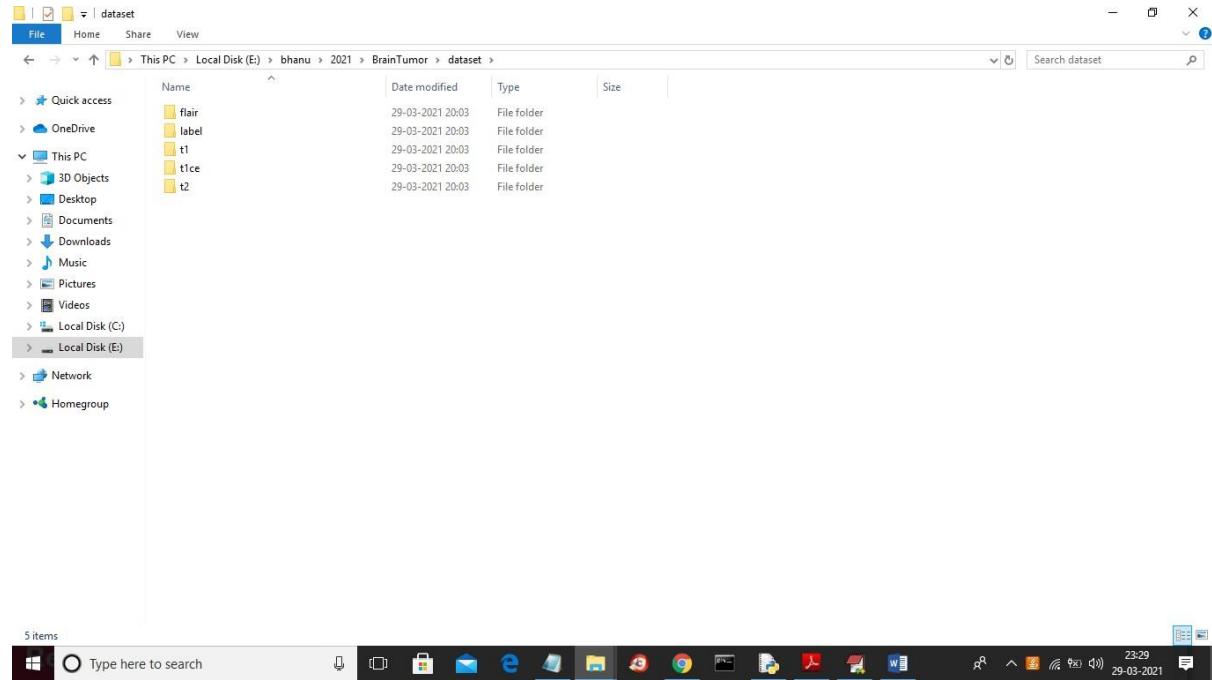
the support of the outside data saved, plus the data flowing from one process to another and the results.



(Fig: 3.3.8 Data flow Diagram)

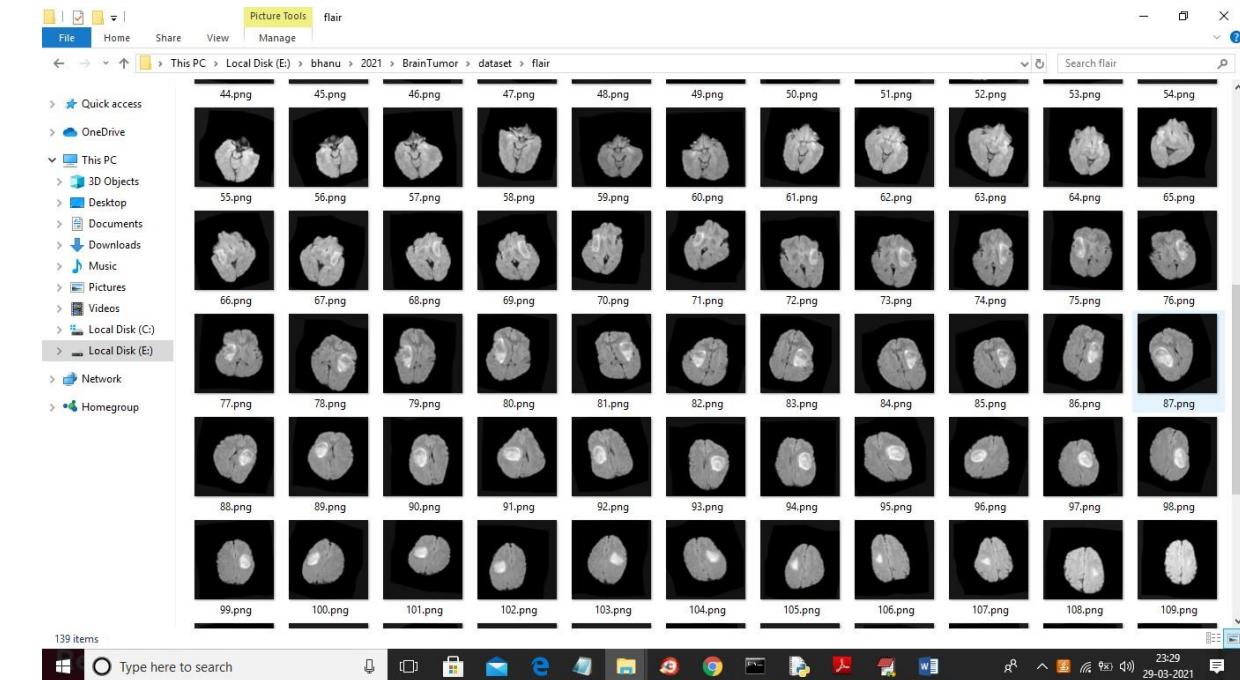
3.4 Project Implementation:

BRATS dataset images are saved inside the dataset folder and in the below screen you can see the dataset content.



(Fig: 3.4.1 Dataset Folder)

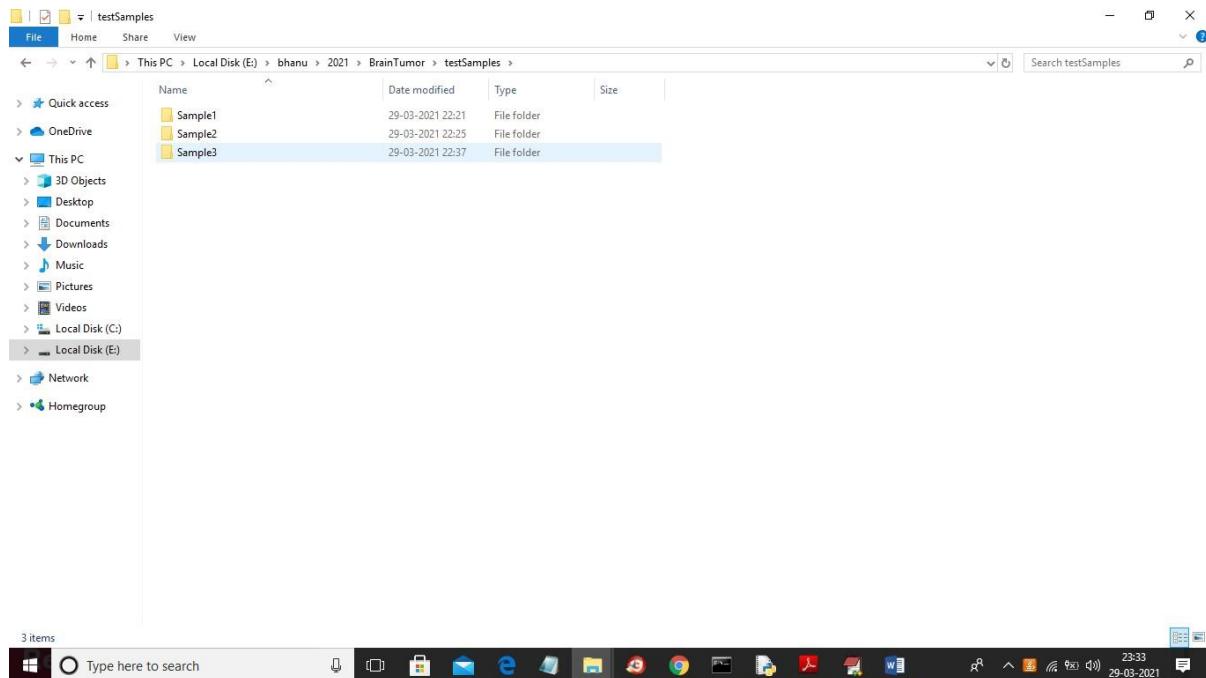
In the above screen we have different format images and you can go inside any folder to see images



(Fig: 3.4.2 Contents of folder)

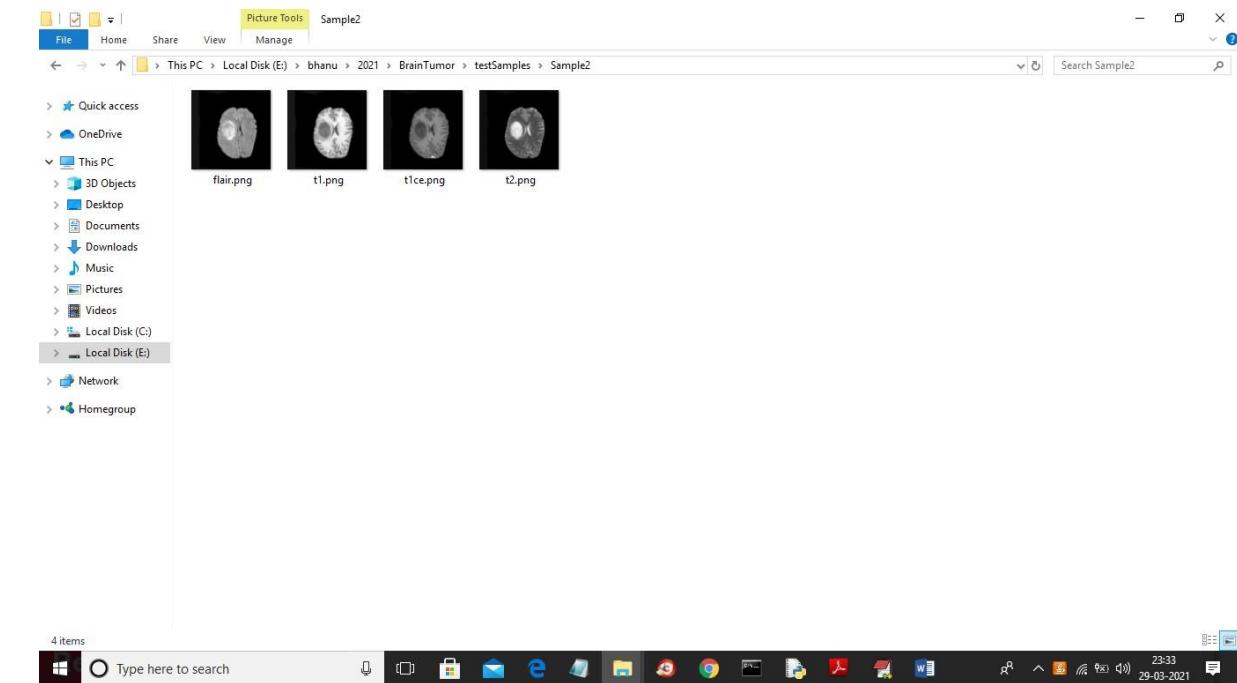
The above dataset is used to train the CNN and UNET model.

After building UNET and CNN models we will upload test images from the ‘test samples’ folder and then the UNET model will give us segmented images. Below screenshots showing the test sample image.



(Fig:3.4.3 Sample Dataset)

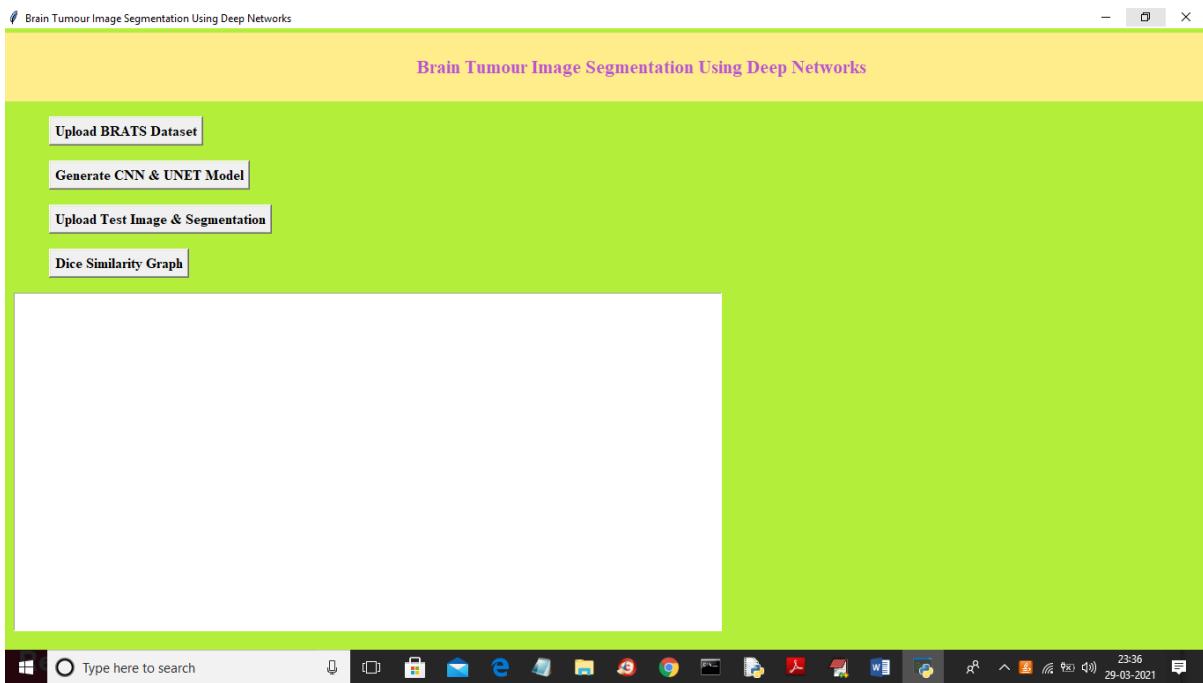
In the above screen we have 3 sample images and now go inside any folder to get the below images.



(Fig:3.4.4 Contents of sample)

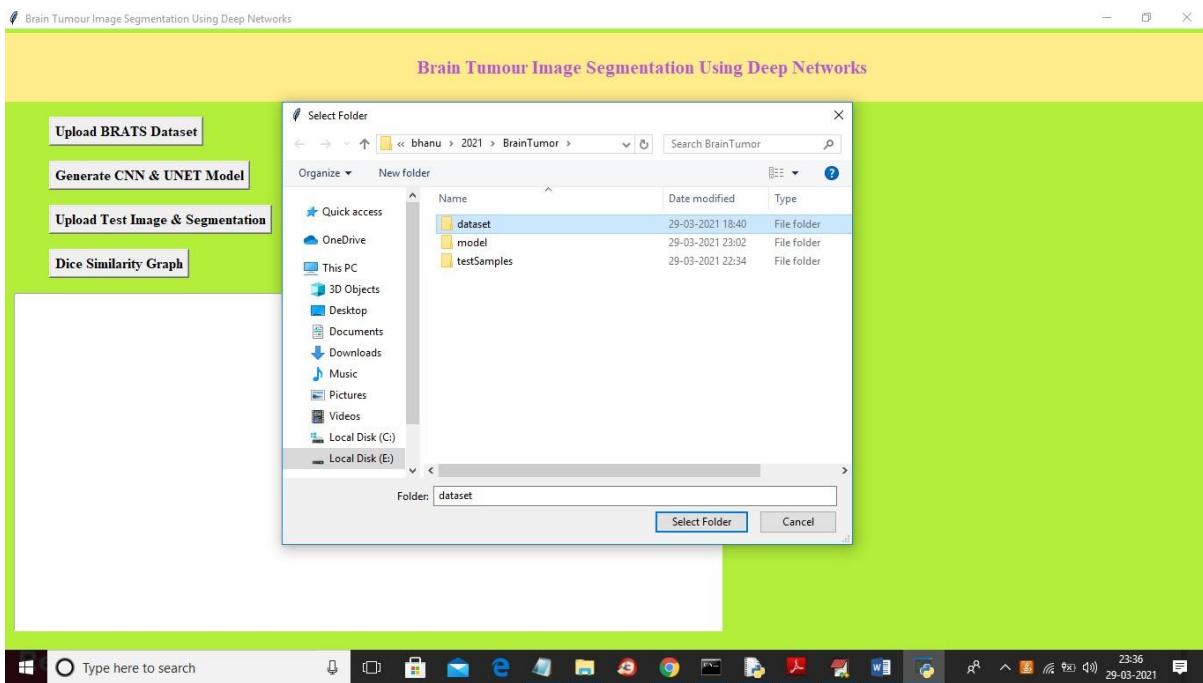
In the above screen, we have FLAIR, T1, TICE and T2 images but we don't have segmented label image after applying model on the above images then we will get segmented label image.

To run the project double click on the 'run.bat' file to get the below screen



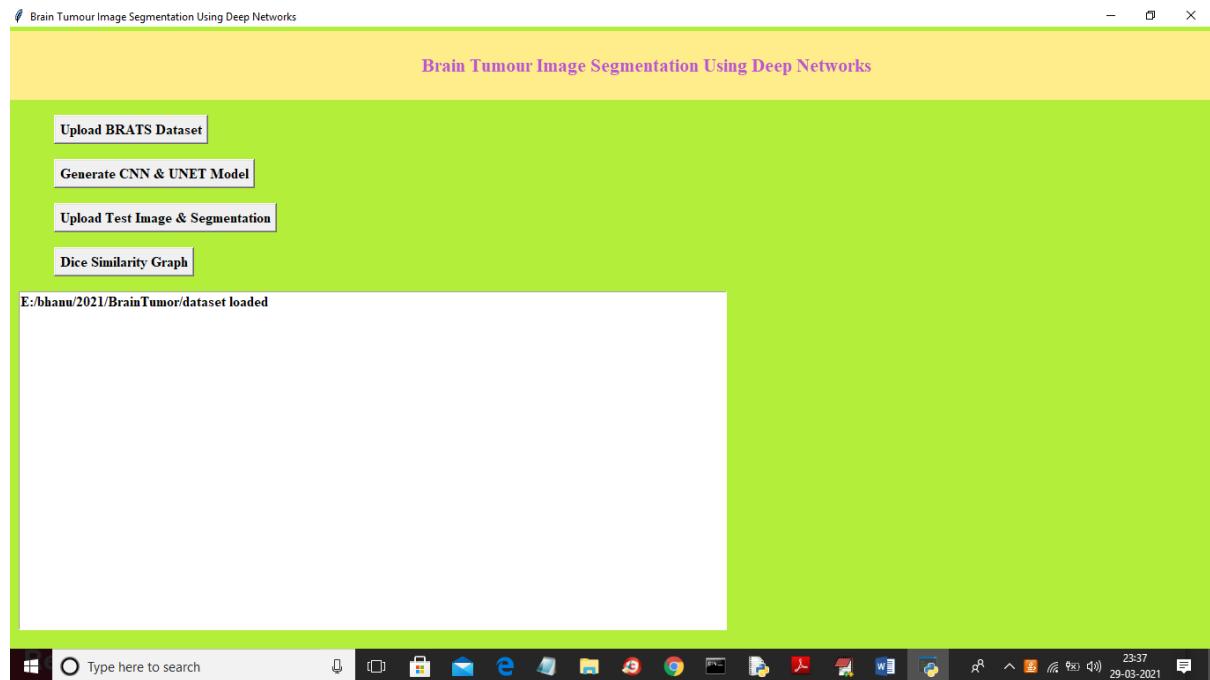
(Fig:3.4.5 User Interface)

In the above screen click on the 'Upload BRATS Dataset' button to upload a dataset



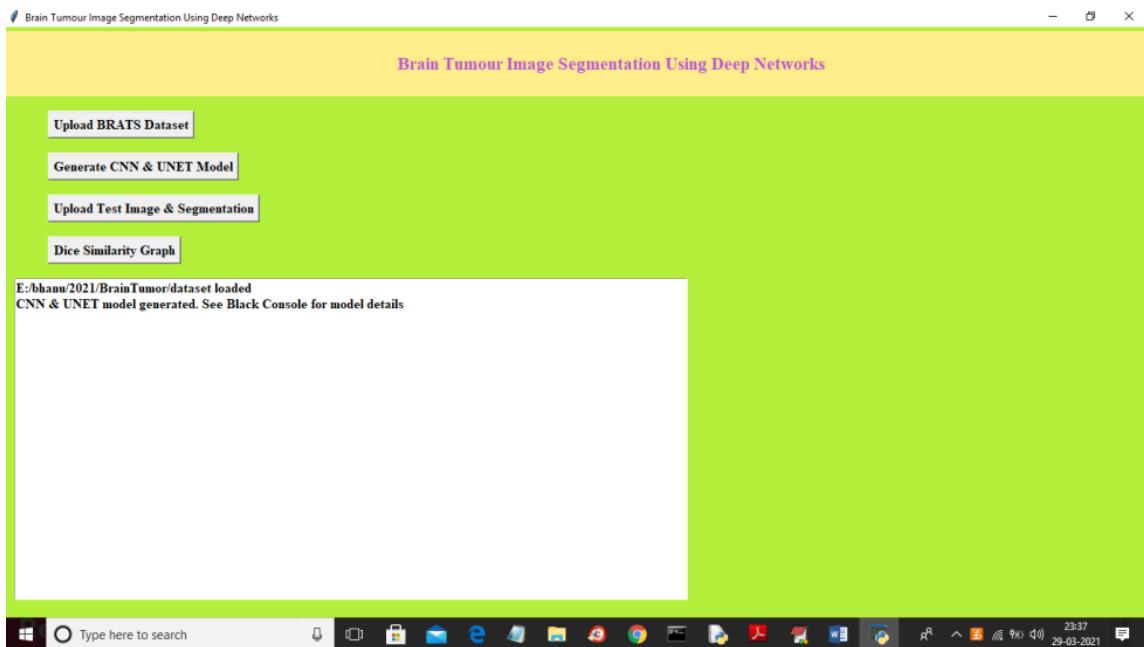
(Fig:3.4.6 Uploading Dataset)

In above screen selecting and uploading ‘dataset’ folder and then click on ‘Select Folder’ button to load dataset and to get below screen



(Fig:3.4.7 Dataset Uploaded)

In the above screen dataset loaded and now click on the ‘Generate CNN & UNET Model’ button to generate models and to get the below screen



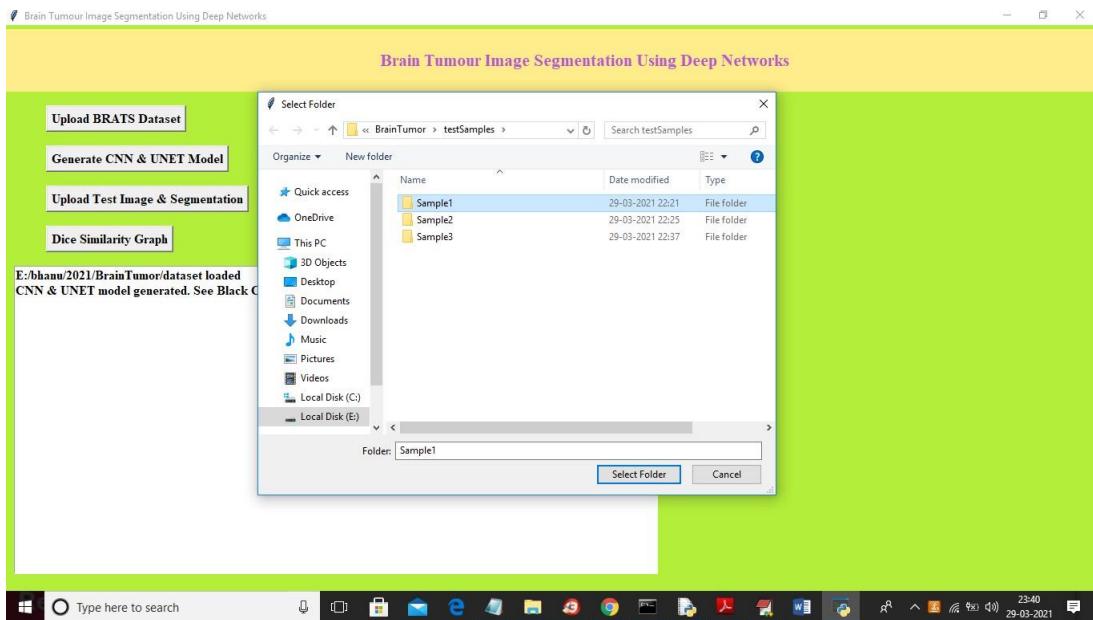
(Fig:3.4.8 Generating Model)

In the above screen we can see both models are generated and we can see below the black console to see CNN and UNET layer details

```
C:\Windows\system32\cmd.exe
E:\bhanu\2021\BrainTumor>python BrainTumour.py
Using TensorFlow backend.
Model: "model_1"
Layer (type)      Output Shape     Param #   Connected to
=====
input_1 (InputLayer)    (None, 64, 64, 1)   0
conv2d_1 (Conv2D)       (None, 64, 64, 32)  320      input_1[0][0]
conv2d_2 (Conv2D)       (None, 64, 64, 32)  9248     conv2d_1[0][0]
max_pooling2d_1 (MaxPooling2D) (None, 32, 32, 32) 0      conv2d_2[0][0]
conv2d_3 (Conv2D)       (None, 32, 32, 64)  18496    max_pooling2d_1[0][0]
conv2d_4 (Conv2D)       (None, 32, 32, 64)  36928    conv2d_3[0][0]
max_pooling2d_2 (MaxPooling2D) (None, 16, 16, 64) 0      conv2d_4[0][0]
conv2d_5 (Conv2D)       (None, 16, 16, 128) 73856    max_pooling2d_2[0][0]
conv2d_6 (Conv2D)       (None, 16, 16, 128) 147584   conv2d_5[0][0]
max_pooling2d_3 (MaxPooling2D) (None, 8, 8, 128) 0      conv2d_6[0][0]
conv2d_7 (Conv2D)       (None, 8, 8, 256)  295168   max_pooling2d_3[0][0]
conv2d_8 (Conv2D)       (None, 8, 8, 256)  590080   conv2d_7[0][0]
max_pooling2d_4 (MaxPooling2D) (None, 4, 4, 256) 0      conv2d_8[0][0]
conv2d_9 (Conv2D)       (None, 4, 4, 512)  1180160   max_pooling2d_4[0][0]
conv2d_10 (Conv2D)      (None, 4, 4, 512)  2359808  conv2d_9[0][0]
conv2d_transpose_1 (Conv2DTranspose) (None, 8, 8, 256) 524544  conv2d_10[0][0]
concatenate_1 (Concatenate) (None, 8, 8, 512) 0      conv2d_transpose_1[0][0]
                                         conv2d_8[0][0]
conv2d_11 (Conv2D)      (None, 8, 8, 256)  1179904   concatenate_1[0][0]
```

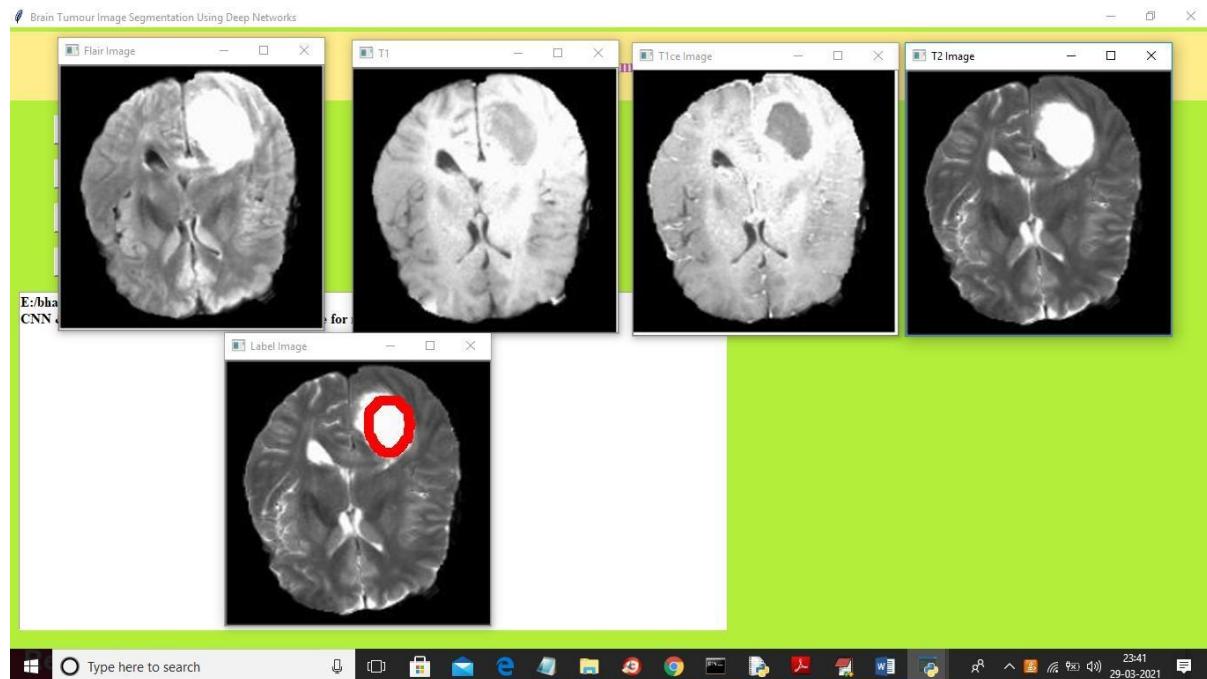
(Fig:3.4.9 Output, Shape, Params)

In the above screen, we can see models are using different size images to filter them and to get the best features from it to build an efficient model now model is generated and now click on the ‘Upload Test Image & Segmentation’ button and then upload test samples to get segmented output



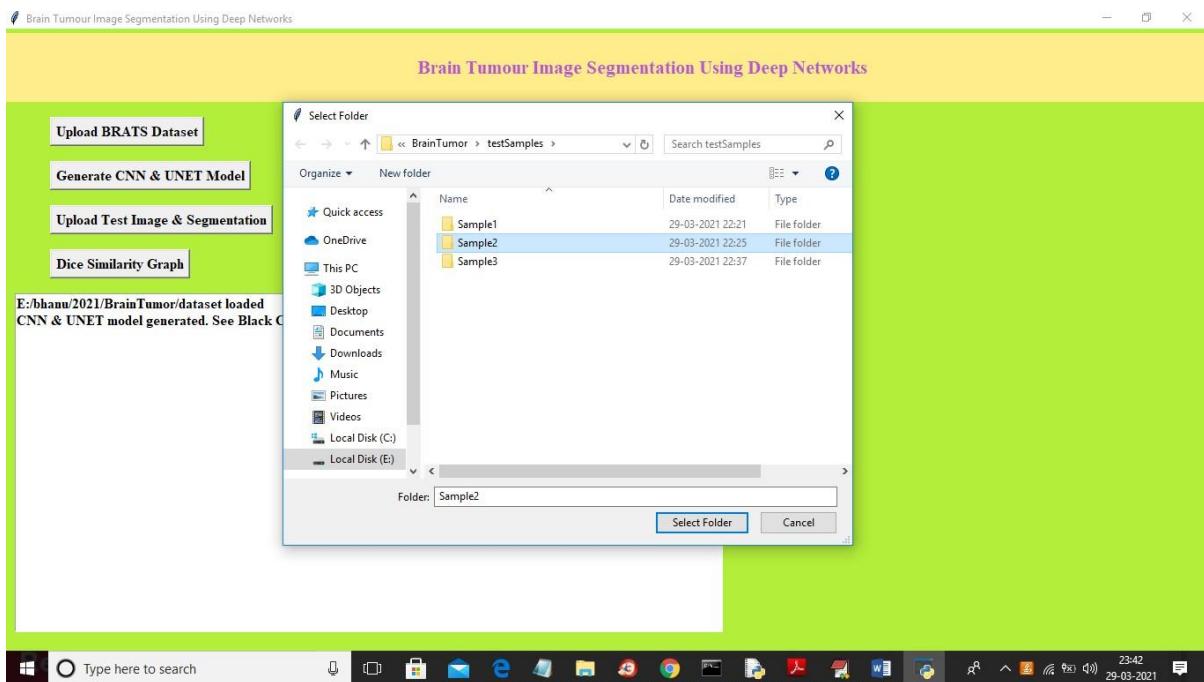
(Fig:3.4.10 Uploading Sample Dataset)

In above screen selecting and uploading ‘Sample1’ folder and then click on ‘Select Folder’ button to get below output.



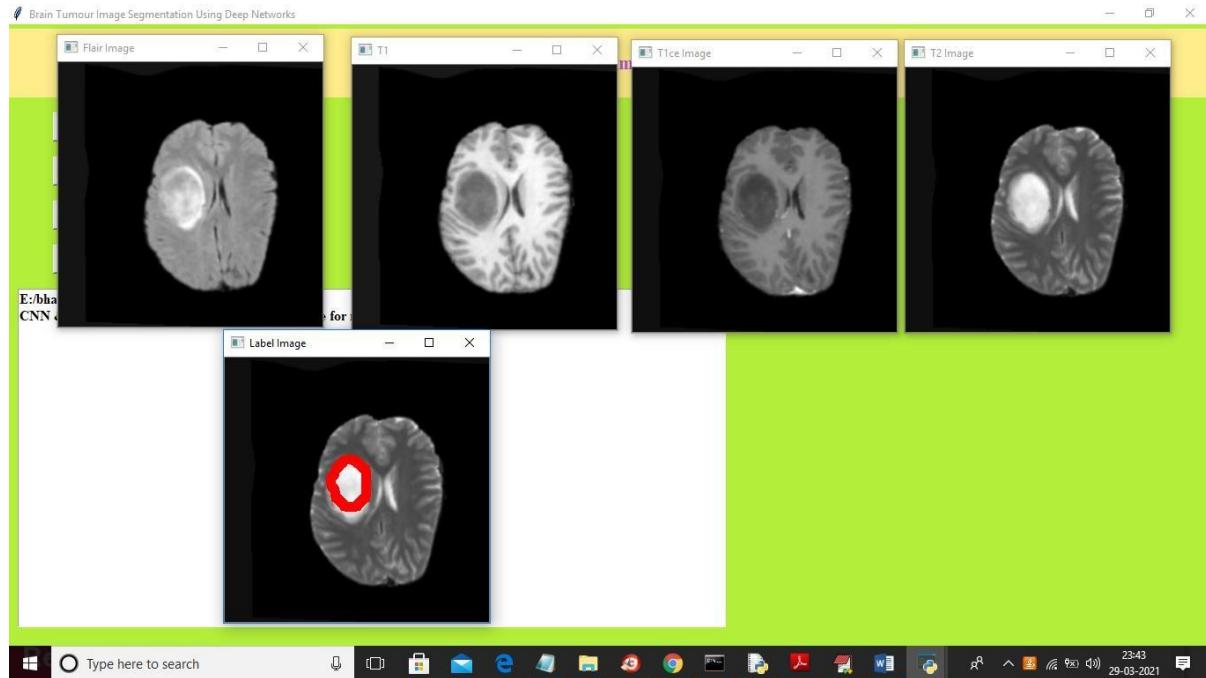
(Fig:3.4.11 Segmentation of MRI Scans)

On the screen above, you can see the top 4 images which are the input images - FLAIR, T1, T2 and T1CE. The 5th image is the predicted image, with the segmented part shown in red. This algorithm accurately detects and marks the tumor area. Now, let's test it with another image.



(Fig:3.4.12 Uploading Sample Dataset-2)

In above screen I am selecting and uploading the ‘Sample2’ folder and then click on ‘Select Folder’ button to load images and to get below output



(Fig:3.4.13 Segmentation of Sample Dataset-2)

CHAPTER 4

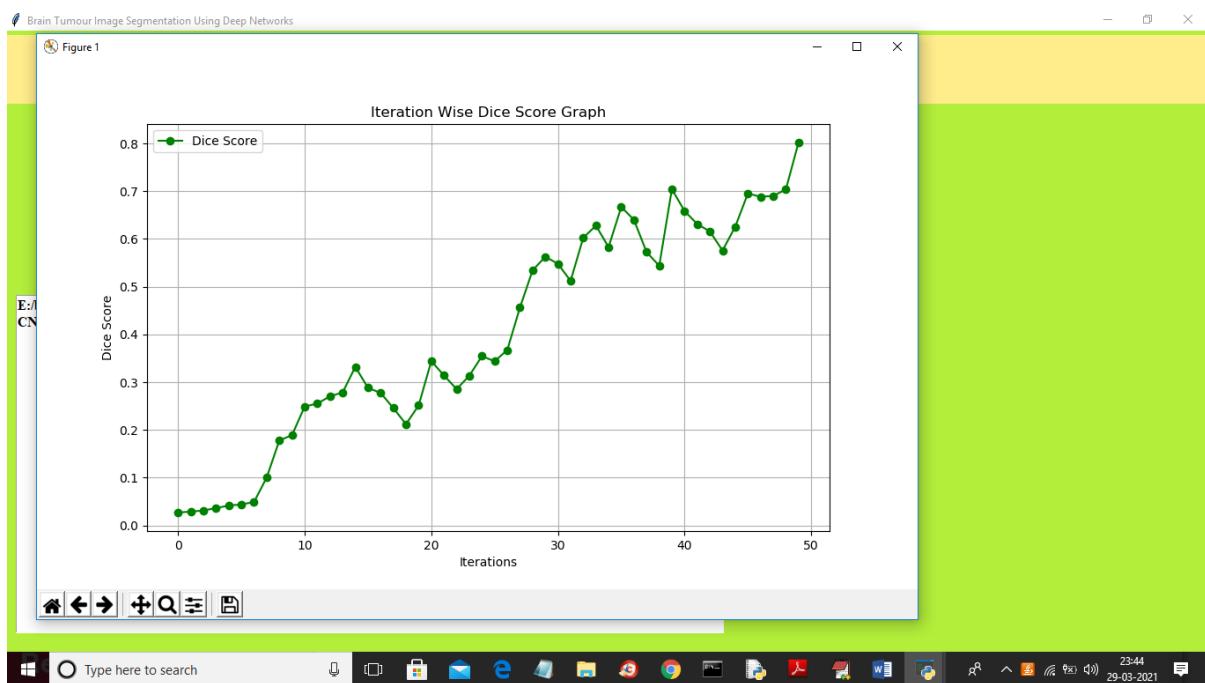
RESULTS AND DISCUSSION

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Results Obtained

On the screen above, you can see four images displayed as input. The fifth image shows the predicted label image with segmented parts around the tumor area. To view the Dice Similarity Graph, click on the 'Dice Similarity Graph' button below.



(Fig:4.1.1 Dice Score Graph)

We built a CNN and UNET model with 50 epochs or iterations. At each iteration, the DICE score between the training and testing images improved, resulting in a final DICE score of 80% (0.8×100). In the graph above, the x-axis represents the epoch, and the y-axis represents the DICE score.

CHAPTER 5

CONCLUSION

CHAPTER 5

CONCLUSION

Conclusions:

Automation and Efficiency:

The automated segmentation approach significantly streamlines the process of identifying and delineating tumor regions in MRI scans. This reduces the reliance on manual segmentation, which can be time-consuming and susceptible to human error.

Clinical Utility:

The U-Net-based segmentation method holds great promise in a clinical setting. It provides medical professionals with a powerful tool for accurate and efficient diagnosis and monitoring of brain tumors.

Potential for Early Detection:

Early detection of brain tumors is crucial for timely medical interventions. The automated segmentation method enhances the potential for early detection, which can lead to improved patient outcomes.

Scalability and Adaptability:

The deep learning-based approach employed in this project offers scalability and adaptability. With additional data and appropriate fine-tuning, the model can potentially be extended to address other types of tumors or even applied to different medical imaging tasks.

Ethical Considerations:

The project places significant emphasis on ethical considerations, including patient privacy, data security, and obtaining informed consent. These aspects are critical in maintaining the highest standards of patient care and research integrity.

CHAPTER 6

REFERENCES

REFERENCES

- [1] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), 234-241.
- [2] Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42, 60-88.
- [3] Greenspan, H., van Ginneken, B., & Summers, R. M. (2016). Guest editorial deep
- [4] Wu, B., et al. (2024). Interactive medical image annotation using improved Attention U-net with compound geodesic distance. *Expert Systems with Applications*, 237, 121282.
- [5] Nguyen, P. X., et al. (2019). Medical image segmentation with stochastic aggregated loss in a unified u-net. In 2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI) (pp. 1-4). IEEE.
- [6] Huang, C., et al. (2019). 3D U 2-Net: A 3D universal U-Net for multi-domain medical image segmentation. In International Conference on Medical Image Computing and Computer-Assisted Intervention 2019 (pp. 291-299). Cham: Springer International Publishing.
- [7] Lu, H., et al. (2022). Half-UNet: A simplified U-Net architecture for medical image segmentation. *Frontiers in Neuroinformatics*, 16, 911679.
- [8] Arora, A., et al. (2021). Brain tumor segmentation of MRI images using processed image driven u-net architecture. *Computers*, 10(11), 139.
- [9] Zahra, E., et al. (2020). Medical image segmentation using a U-Net type of architecture. arXiv preprint arXiv:2005.05218.
- [10] Zhuang, J. (2018). LadderNet: Multi-path networks based on U-Net for medical image segmentation. arXiv preprint arXiv:1810.07810.
- [11] Ali, O., et al. (2022). Implementation of a modified U-Net for medical image segmentation on edge devices. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 69(11), 4593-4597.
- [12] Isensee, F., et al. (2018). nnu-net: Self-adapting framework for u-net-based medical image segmentation. arXiv preprint arXiv:1809.10486.
- [13] Hazarika, R. A., et al. (2022). Hippocampus segmentation using U-net convolutional network from brain magnetic resonance imaging (MRI). *Journal of Digital Imaging*, 35(4), 893-909.

- [14] Sun, J., et al. (2020). Saunet: Shape attentive u-net for interpretable medical image segmentation. In Medical Image Computing and Computer Assisted Intervention—MICCAI 2020 (pp. 797-806). Springer International Publishing.
- [15] Siddique, N., et al. (2020). U-Net and its variants for medical image segmentation: theory and applications. arXiv preprint arXiv:2011.01118.
- [16] Du, G., et al. (2020). Medical image segmentation based on u-net: A review. *Journal of Imaging Science and Technology*, 64(5), 50301.
- [17] Siddique, N., et al. (2021). U-net and its variants for medical image segmentation: A review of theory and applications. *IEEE Access*, 9, 82031-82057.
- [18] Alom, M. Z., et al. (2019). Recurrent residual U-Net for medical image segmentation. *Journal of Medical Imaging*, 6(1), 014006.
- [19] Wei, J., et al. (2021). Genetic U-Net: automatically designed deep networks for retinal vessel segmentation using a genetic algorithm. *IEEE Transactions on Medical Imaging*, 41(2), 292-307.
- [19] Yin, X. X., et al. (2022). U-Net-Based medical image segmentation. *Journal of Healthcare Engineering*, 2022.
- [20] Zheng, P., et al. (2022). Brain tumour segmentation based on an improved U-Net. *BMC Medical Imaging*, 22(1), 1-9.
- [21] Chetty, G., et al. (2022). A low resource 3D U-Net based deep learning model for medical image analysis. *International Journal of Information Technology*, 14(1), 95-103.
- [22] Kermi, A., et al. (2019). Deep convolutional neural networks using U-Net for automatic brain tumor segmentation in multimodal MRI volumes. In *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries* (pp. 37-48). Springer International Publishing.
- [23] Chen, W., et al. (2022). Deep learning-based medical image segmentation of the aorta using XR-MSF-U-Net. *Computer Methods and Programs in Biomedicine*, 225, 107073.
- [24] Jwaid, W. M., et al. (2021). DEVELOPMENT OF BRAIN TUMOR SEGMENTATION OF MAGNETIC RESONANCE IMAGING (MRI) USING U-NET DEEP LEARNING. *Eastern-European Journal of Enterprise Technologies*, 112.
- [25] Deng, Y., et al. (2022). ELU-net: an efficient and lightweight U-net for medical image segmentation. *IEEE Access*, 10, 35932-35941.

CHAPTER 7

APPENDIX

7.1 Project Code:

Importing Libraries

```
from tkinter import messagebox
from tkinter import *
from tkinter import simpledialog
import tkinter
from tkinter import simpledialog
from tkinter import filedialog
import numpy as np
from tkinter.filedialog import askopenfilename
import pickle
import os
import cv2
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split

from keras.models import *
from keras.layers import * from
keras.optimizers import *

gui = tkinter.Tk()
gui.title("Brain Tumour Image Segmentation Using Deep Networks")
gui.geometry("1300x1200")

global filename
global model
global X, Y
```

Calculating Dice Coefficient

```
def dice_coef(y_true, y_pred):
    y_true_f = keras.flatten(y_true)
    y_pred_f = keras.flatten(y_pred)
    intersection = keras.sum(y_true_f * y_pred_f)
    return (2. * intersection + 1) / (keras.sum(y_true_f) + keras.sum(y_pred_f) + 1)
```

```
def dice_coef_loss(y_true, y_pred):
    return -dice_coef(y_true, y_pred)
```

```
def getModel(input_size=(64,64,1)):
    inputs = Input(input_size)

    conv1 = Conv2D(32, (3, 3), activation='relu', padding='same')(inputs)
    conv1 = Conv2D(32, (3, 3), activation='relu', padding='same')(conv1)
    pool1 = MaxPooling2D(pool_size=(2, 2))(conv1)

    conv2 = Conv2D(64, (3, 3), activation='relu', padding='same')(pool1)
```

```
conv2 = Conv2D(64, (3, 3), activation='relu', padding='same')(conv2)
pool2 = MaxPooling2D(pool_size=(2, 2))(conv2)

conv3 = Conv2D(128, (3, 3), activation='relu', padding='same')(pool2)
conv3 = Conv2D(128, (3, 3), activation='relu', padding='same')(conv3)
pool3 = MaxPooling2D(pool_size=(2, 2))(conv3)

conv4 = Conv2D(256, (3, 3), activation='relu', padding='same')(pool3)
conv4 = Conv2D(256, (3, 3), activation='relu', padding='same')(conv4)
pool4 = MaxPooling2D(pool_size=(2, 2))(conv4)

conv5 = Conv2D(512, (3, 3), activation='relu', padding='same')(pool4)
conv5 = Conv2D(512, (3, 3), activation='relu', padding='same')(conv5)

up6 = concatenate([Conv2DTranspose(256, (2, 2), strides=(2, 2), padding='same')(conv5), conv4],
axis=3)
conv6 = Conv2D(256, (3, 3), activation='relu', padding='same')(up6)
conv6 = Conv2D(256, (3, 3), activation='relu', padding='same')(conv6)

up7 = concatenate([Conv2DTranspose(128, (2, 2), strides=(2, 2), padding='same')(conv6), conv3],
axis=3)
conv7 = Conv2D(128, (3, 3), activation='relu', padding='same')(up7)
conv7 = Conv2D(128, (3, 3), activation='relu', padding='same')(conv7)

up8 = concatenate([Conv2DTranspose(64, (2, 2), strides=(2, 2), padding='same')(conv7), conv2],
axis=3)
conv8 = Conv2D(64, (3, 3), activation='relu', padding='same')(up8)
conv8 = Conv2D(64, (3, 3), activation='relu', padding='same')(conv8)

up9 = concatenate([Conv2DTranspose(32, (2, 2), strides=(2, 2), padding='same')(conv8), conv1],
axis=3)
conv9 = Conv2D(32, (3, 3), activation='relu', padding='same')(up9)
conv9 = Conv2D(32, (3, 3), activation='relu', padding='same')(conv9)

conv10 = Conv2D(1, (1, 1), activation='sigmoid')(conv9)

return Model(inputs=[inputs], outputs=[conv10])
```

Uploading the dataset

```
def uploadDataset():
    global X, Y global
    filename
    text.delete('1.0', END)
    filename = filedialog.askdirectory(initialdir=".")
```

```
text.insert(END,filename+" loaded\n");
```

```
X = []
```

```
Y = []
for root, dirs, directory in os.walk(filename):
    for i in range(len(directory)):
        img = cv2.imread(train_directory+"/"+directory[i],0)
        img = cv2.resize(img,(64,64), interpolation = cv2.INTER_CUBIC)
        X.append(img)
        img = cv2.imread("dataset/label/"+directory[i],0)
        img = cv2.resize(img,(64,64), interpolation = cv2.INTER_CUBIC)
        Y.append(img)

X = np.asarray(X)
Y = np.asarray(Y)
```

Generating the model

```
def generateModel():
    global model
    global X, Y
    dim = 64
    X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size = 0.10, random_state = 1)
    X_train = X_train.reshape(len(X_train),dim,dim,1)
    y_train = y_train.reshape(len(y_train),dim,dim,1)
    X_test = X_test.reshape(len(X_test),dim,dim,1)
    y_test = y_test.reshape(len(y_test),dim,dim,1)
    images = np.concatenate((X_train,X_test),axis=0)
    mask = np.concatenate((y_train,y_test),axis=0)
    tr = X_train[12]
    yr = y_train[12]
    cv2.imshow('tr',tr)
    cv2.imshow('yr',yr)
    cv2.waitKey(0)
    model = get_model(input_size=(64,64,1))
    with open('model/model.json', "r") as json_file:
        loaded_model_json = json_file.read()
        model = model_from_json(loaded_model_json)
    json_file.close()
    model.load_weights("model/model_weights.h5")
    model._make_predict_function()
    print(model.summary())
    text.insert(END,"CNN & UNET model generated. See Black Console for model details\n")
    model.compile(optimizer=Adam(lr=1e-5), loss=dice_coef_loss, metrics=[dice_coef,
    'binary_accuracy'])
    print(model.summary())
    model.compile(optimizer=Adam(lr=2e-4), loss=[dice_coef_loss], metrics = [dice_coef,
    'binary_accuracy'])
    train_vol, validation_vol, train_seg, validation_seg = train_test_split((images-127.0)/127.0,
    (mask>127).astype(np.float32),
    test_size = 0.1,random_state = 2018)
    train_vol, test_vol, train_seg, test_seg = train_test_split(train_vol,train_seg,
```

```
        test_size = 0.1,  
        random_state = 2018)  
hist = model.fit(x = train_vol, y = train_seg, batch_size = 16, epochs = 50, validation_data  
=(test_vol,test_seg))  
model.save_weights('model/model_weights.h5')  
model_json = model.to_json()  
with open("model/model.json", "w") as json_file:  
    json_file.write(model_json)  
f = open('model/history.pkl', 'wb')  
pickle.dump(hist.history, f) f.close()
```

Segmentation task

```
def getSegmentation():  
    img = cv2.imread('myimg.png')  
    orig = cv2.imread('test1.png')  
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)  
    thresh = cv2.threshold(gray, 30, 255, cv2.THRESH_BINARY)[1]  
    contours = cv2.findContours(thresh, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)  
    contours = contours[0] if len(contours) == 2 else contours[1]  
    min_area = 0.95*180*35  
    max_area = 1.05*180*35  
    result = orig.copy()  
    for c in contours:  
        area = cv2.contourArea(c)  
        cv2.drawContours(result, [c], -1, (0, 0, 255), 10)if  
        area > min_area and area < max_area:  
            cv2.drawContours(result, [c], -1, (0, 255, 255), 10)  
    return result
```

```
def TumourSegmentation():  
    global model  
    filename = filedialog.askdirectory(initialdir="testSamples")  
    img = cv2.imread(str(filename)+'/t2.png',0)  
    img = cv2.resize(img,(64,64), interpolation = cv2.INTER_CUBIC)  
    img = img.reshape(1,64,64,1)  
    img = (img-127.0)/127.0  
    preds = model.predict(img)  
    preds = preds[0]  
    print(preds.shape)  
    orig = cv2.imread(str(filename)+'/t2.png',0)  
    orig = cv2.resize(orig,(300,300),interpolation = cv2.INTER_CUBIC)
```

```
cv2.imwrite("test1.png",orig)  
  
flair = cv2.imread(str(filename)+'/flair.png',0)  
flair = cv2.resize(flair,(300,300),interpolation = cv2.INTER_CUBIC)t1  
= cv2.imread(str(filename)+'/t1.png',0)  
t1 = cv2.resize(t1,(300,300),interpolation = cv2.INTER_CUBIC)  
t1ce = cv2.imread(str(filename)+'/t1ce.png',0)  
t1ce = cv2.resize(t1ce,(300,300),interpolation = cv2.INTER_CUBIC)
```

```
preds = cv2.resize(preds,(300,300),interpolation = cv2.INTER_CUBIC)
cv2.imwrite("myimg.png",preds*255)
preds = getSegmentation()
cv2.imshow('Flair Image',flair)
cv2.imshow('T1',t1)
cv2.imshow("T1ce Image",t1ce)
cv2.imshow("T2 Image",orig)
cv2.imshow("Label Image",preds)
cv2.waitKey(0)
```

Generating the graph for Dice Coefficient

```
def graph():
    f = open('model/history.pckl', 'rb')
    data = pickle.load(f)
    f.close()
    dice = data['dice_coef'] for i
    in range(len(dice)): dice[i]
        = dice[i] * 2
    plt.figure(figsize=(10,6))
    plt.grid(True) plt.xlabel('Iterations')
    plt.ylabel('Dice Score')
    plt.plot(dice, 'ro-', color = 'green')
    plt.legend(['Dice Score'], loc='upper left')
    #plt.xticks(wordloss.index)
    plt.title('Iteration Wise Dice Score Graph')
    plt.show()
```

Making UI with tkinter

```
font = ('times', 16, 'bold')
title = Label(gui, text='Brain Tumour Image Segmentation Using Deep Networks')
title.config(bg='LightGoldenrod1', fg='medium orchid')
title.config(font=font)
title.config(height=3, width=120)
title.place(x=0,y=5)
```

```
font1 = ('times', 12, 'bold')
text=Text(gui,height=20,width=100)
scroll=Scrollbar(text)
text.configure(yscrollcommand=scroll.set)
text.place(x=10,y=300)

text.config(font=font1)

font1 = ('times', 12, 'bold')
```

```
loadButton = Button(gui, text="Upload BRATS Dataset", command=uploadDataset)
loadButton.place(x=50,y=100)
loadButton.config(font=font1)

uploadButton = Button(gui, text="Generate CNN & UNET Model", command=generateModel)
uploadButton.place(x=50,y=150)
uploadButton.config(font=font1)

descButton = Button(gui, text="Upload Test Image & Segmentation",
command=TumourSegmentation)
descButton.place(x=50,y=200)
descButton.config(font=font1)

closeButton = Button(gui, text="Dice Similarity Graph", command=graph)
closeButton.place(x=50,y=250)
closeButton.config(font=font1)

gui.config(bg='OliveDrab2')
gui.mainloop()
```

8 GITHUB LINK

<https://github.com/akshay-1209/Medical-Image-Segmentation-For-Brain-Tumor-Detection-Using-U-net-s>

9 DOI

<https://doi.org/10.22214/ijraset.2024.59327>

10 PUBLISHED PAPER



Volume: 12 **Issue:** III **Month of publication:** March 2024

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com



Medical Image Segmentation for Brain Tumor Detection Using U-NET

Jarpula Sivakumar¹, Akshay Tonde², Vuyyuru Namitha³, S. Sangeetha⁴

^{1, 2, 3}Department of Computer Science and Engineering, CMR College of Engineering & Technology, Hyderabad, India

⁴Assistant Professor, Department of Computer Science and Engineering, CMR College of Engineering & Technology, Hyderabad, India

Abstract: Brain tumors encompass a wide variety of growths that can occur in the cranial cavity. These tumors can be benign or malignant, and early identification is important for suitable treatment. MRI is a non-invasive imaging method that provides detailed anatomical information and is useful in diagnosing and monitoring brain tumors. However, identifying tumors from complex MRI data remains a difficult and challenging task. This project uses image segmentation techniques to identify and identify tumors on MRI scans. The technology works on the principles of deep learning, an intelligence process that enables computers to learn and recognize patterns in large data sets. This learning process helps doctors make informed decisions by distinguishing between healthy brain tissue and tumor growth. The unique design of the U-Net architecture, characterized by a U-shaped pattern with compactness and detail, makes it possible to capture complex patterns and fine details in medical images. Using different data from many brain MRI scans, the U-Net algorithm was trained to depict tumor regions with pixel-level accuracy.

Keywords: Brain Tumor Detection, Medical Image Segmentation, U-Net, CNN, Deep Learning.

I. INTRODUCTION

Precise tumor segmentation in medical photographs has great significance since it provides vital information required for cancer analysis, diagnosis, treatment planning and tracking the course of the illness. Based on their origin, brain tumors—which are among the worst malignancies worldwide—are divided into primary and secondary forms. Of all brain tumors that are malignant, gliomas, the most similar type, are derived from brain glial cells and include around 80% of cases. Gliomas can present as aggressive high-grade gliomas (HGG) or glioblastomas, which require rapid intervention, or as slowly developing low-grade (LGG) tumors with a generally good prognosis. Glioblastoma patients, in spite of the best medical and surgical care available, have a median survival of only around 14 months, and their 5-year survival rate is essentially nonexistent. Therefore, prompt diagnosis is essential to successful treatment. The recommended method for evaluating brain tumors is magnetic resonance imaging (MRI), which offers a variety of 3D MRI modalities including fluid-attenuated inversion recovery (FLAIR), T2-weighted, post-contrast T1-weighted (T1ce), and T1-weighted. These modalities reveal distinct subregions of tumors, with FLAIR and T2 highlighting the entire tumor, including infiltrative edema, while T1 and T1ce images depict the tumor core, excluding peritumoral edema. Integrating information from these scans aids in detecting various tumor subregions. A platform for assessing machine learning models targeted at tumor segmentation is offered by the Multimedia Brain Tumor Segmentation Challenge (BraTS). It provides expert-annotated ground truths and a dataset of 3 Dimensional MRI pictures of gliomas (for both LGG and HGG). Automated segmentation of gliomas from multichannel MRI images facilitates precise, repeatable tumor analysis and expedites surgical planning and diagnosis. While unattended studying algorithms and CNNs have become state-of-the-art techniques due to their scope to automatically learn pertinent features, traditional methods entail feature engineering. However, tumor segmentation remains challenging due to tumor heterogeneity in shape, size, appearance, and the ambiguous boundary between cancerous and healthy tissue, compounded by intensity variability in MRI data. Multiple 3D CNN models are used in this work to segment brain tumors from multimodal MRI data. The probability maps of these models as an ensemble improve forecast stability. With optimal hyperparameters, each network is trained independently using the dataset from the 2019 BraTS challenge. The system obtained dice scores of 0.846, 0.906, and 0.750 for the tumor core, total tumor, and enhancing tumor, respectively.

II. OBJECTIVE OF THE PROJECT

In order to analyze diseases and track their course, brain tumors using multimodal MR images must be automatically segmented. Because gliomas are heterogeneous and malignant, it is critical to use precise and effective segmentation methods to divide tumors



into intra-tumoral classes. When it comes to tasks like semantic segmentation, deep learning algorithms have proven to perform better than conventional context based on computer perception techniques. With their widespread application in bio-genetic image segmentation, convolutional neural CNNs have made tremendous progress toward the state-of-the-art accuracy for brain tumor separation. In this work, we provide an ensemble process that combines a U-Net and a 3 Dimensional CNN for segmentation. This simple yet effective combination strategy produces better and more precise forecasts. The BraTS-19 challenge dataset was used for each model's independent training and evaluation, which produced segmentation maps with significantly different segmented tumor sub-regions. The final prediction was then produced by variably ensembleing these diverse results. On the validation set, our suggested ensemble obtained accuracy values of 0.750, 0.906, and 0.846 for the detected tumor, complete tumor, and tumor core, respectively. These findings show good performance when compared to current cutting-edge architectures.

III. LITERATURE SURVEY

A. Clinical image examination using MRI for brain tumor research is gaining increasing attention due to the required to evaluate more data efficiently and objectively

Although the first technology for the brain tumor imaging emerged nearly two decades ago, the modern technique has grown and approached modern medicine. This system aims to supply an overview of brain tumors and imaging, starting with a brief introduction. It then examines state-of-the-art segmentation, registration, and modeling techniques for brain imaging of tumors, focusing on gliomas. Segmentation aims to delineate the tumor, its subdivisions and surrounding tissue, while recording and modeling the problems caused by tumor-induced morphological changes. This review examines various methods, considering those suitable for utilize in clinical imaging protocols. Finally, it evaluates the current situation and predicts future developments and trends, paying particular attention to recent uses in radiological evaluation techniques.

B. There are notable national variations in the worldwide event of malignant brain and other CNS tumors, according to research

The objective of this research is to evaluate age and histology changes while estimating distinct histology trends in various parts of the world. The age-adjusted incidence rates (AAIR) per 100,000 years for brain tumors and other CNS malignancies were determined in this study using data from the International Agency for Research on Cancer (IARC) and the Central Brain Tumor Registry of the United States (CBTRUS).

Has made use of. Five countries with cancer. These tumors add to the global burden of cancer-related morbidity and mortality. Variations in cancer incidence among locations may be due to environmental or genetic variables, which will further our knowledge of the disease. For academics, clinicians, illness support groups, and collaborative research, a deeper grasp of the worldwide frequency of brain tumors is crucial.

C. The most extensive registry system in the country is maintained by the Central Brain Tumor Registry of the United States (CBTRUS), which collaborates with the National Cancer Institute (NCI) and the Centers for Disease Control (CDC)

For cancers of the central nervous system, including brain tumors. By presenting the most recent population-based data on brain tumors, both malignant and non-malignant, this publication offers accuracy and updates. Age-adjusted rates of incidence and mortality were found in the US population in 2000. Specifically, meningiomas are the most benign tumors, whereas glioblastoma is the most common brain tumor. Disparities in gender and ethnicity are prevalent, and conditions differ across various groups. The report also sheds light on potential future diagnostics and emphasizes the high death rates linked to brain tumors, underscoring the significance of ongoing research and development efforts to enhance patient outcomes.

D. When compared to the 2007 classification, the World Health Organization's 2016 Central Classification of Tumor Nervous System is a major improvement

This classification, which combines genetic and histological factors to identify tumors, ushers in a new era in the diagnosis of CNS cancer. Diffuse gliomas, medulloblastomas, and other embryonal tumors undergo a major reorganization characterized by novel formations determined by histological and molecular characteristics. The use of soft tissue grafts in the treatment of fibrous tumors/hemopericytomas and the addition of brain invasion as a therapy for atypical meningiomas are two significant modifications. In order to enhance patient outcomes, the classification reform seeks to streamline clinical, experimental, and epidemiological research.



E. When temozolomide was added to radiation therapy for newly diagnosed glioblastoma, survival significantly improved with little to no additional harm

The study indicated that the combination group had a higher median survival and two-year survival when comparing radiation therapy alone with radiation therapy plus temozolomide. The advantages of combination therapy for survival outweigh the dangers, notwithstanding the possibility of hematological toxic effects in a small percentage of patients. This study offers promise for better results for individuals with glioblastoma, a dangerous illness, and emphasizes the value of various medications in its treatment.

IV. SYSTEM ANALYSIS

A. Existing System

To simplify the brain tumor process, the authors took advantage of the popularity of profound understanding in classifying clinical images and presented a new method combining 3D CNN and U-Net algorithms. Expanding on this basis, a combination of two deep learning methods (CNN and U-Net) is used to enhance the segmentation procedure even more. Each algorithm was trained separately using BRATS brain tumor data, and then its predictions were combined, or concatenated, to create the final segmentation. More importantly, the resulting results show that Dice has a high score, indicating the success of the shape's cross-section. The main goal is to expand a brain segmentation method that increases efficiency and accuracy compared to existing methods. However, despite the promises of the approach, some shortcomings must be acknowledged:

- 1) *Lower Accuracy:* Although the plan is promising, lower accuracy compared to expectations is still an issue. Despite the high scores mentioned above, there may be cases where the segmentation fails to capture all tumors or misclassifies healthy tissue as tumor, resulting in incorrect diagnosis, monitoring, and treatment planning.
- 2) *Computational Complexity:* Integrating multiple deep learning algorithms and using hybrid methods will introduce computational complexity, resulting in longer processing times and the need for additional resources. This may make the strategy less scalable, particularly when working with large data or real-time systems that need to adapt quickly.
- 3) *Dependency on Training Data:* The performance of segmentation depends on the quality and representativeness of the training data in the BRATS dataset. Insufficient or incorrect information can cause the quality of the model to decrease, which can lead to poor performance with missing data or different tumor types
- 4) *Sensitivity to Hyperparameters:* Deep learning models such as CNN and U-Net are sensitive to hyperparameters and it can be difficult to tune them for optimal performance. Insufficient hyperparameter changes can lead to unsatisfactory segmentation results and even unstable models that require more testing and computational resources.
- 5) *Interpretation:* Deep learning-based segmentation often lacks interpretation, making it difficult to understand the reasons behind segmentation decisions. This can be a problem for doctors who rely on transparent and explained medical decisions.
- 6) *Limitations:* The proposed method may have general problems for many or missing data outside the BRATS dataset. Changes in imaging techniques, tumor characteristics, or patient populations can make it difficult for a system to be adaptable and generalizable to multiple sites.
- 7) *Overfitting:* There is a risk of overextend, particularly when trading with little or unbalanced data. Rather than learning general features, the model will identify specific patterns for the training data, resulting in decreased operation on unvisible data or new tumors. Control measures may be needed to reduce this risk.

B. Proposed System

We use four different images, FLAIR, T1, T2, and T1CE, and the matching annotated segmented image to carry out this job. These pictures are part of a multi-institutional dataset that was obtained from 19 distinct sources and includes multichannel MRI scans for every patient. Tumor subregion delineation is facilitated by the inclusion of T1, T1 contrast enhanced (T1ce), T2-weighted (T2), and Fluid Redused Inversion Recovery (FLAIR) images in the dataset. Preprocessing techniques, such as skull-stripping methods to eliminate non-brain tissue artifacts, are used to guarantee uniformity throughout the data.

Advantages:

- 1) *Greater Accuracy:* By offering complementing information from various imaging views, the utilize of numerous modalities and a diverse dataset improves the accuracy of tumor segmentation. Tumor delineation accuracy is increased and ambiguity is decreased using this all-encompassing method.
- 2) *Improved Visualization:* Combining various imaging modalities makes it possible to see the surrounding tissues and tumor boundaries more clearly, which helps doctors determine the precise location of tumor margins and gauge the degree of tumor infiltration.



- 3) *Improved Diagnostic Specificity:* By integrating various imaging modalities, the segmentation process can achieve higher diagnostic specificity, enabling the identification of subtle differences in tumor characteristics and facilitating more accurate diagnosis and treatment planning.
- 4) *Robustness to Imaging Artifacts:* Utilizing multiple images mitigates the effect of imaging artifacts and variations in image quality, enhancing the robustness of the segmentation algorithm and ensuring reliable results across different datasets and imaging protocols.
- 5) *Comprehensive Tumor Assessment:* The combination of FLAIR, T1, T2, and T1CE images allows for a comprehensive assessment of tumor morphology, heterogeneity, and vascularity, providing clinicians with valuable insights into tumor behavior and aiding in personalized treatment strategies.
- 6) *Potential for Early Detection:* The integration of multimodal imaging data may enable fast detection of subtle tumor changes or recurrence, facilitating prompt intervention and improving patient outcomes.
- 7) *Research and Medical Utility:* The availability of a diverse and comprehensive dataset enhances the utility of the segmentation model for both research purposes and clinical applications, fostering advancements in brain tumor analysis and treatment optimization.

V. PROPOSED SYSTEM ARCHITECTURE

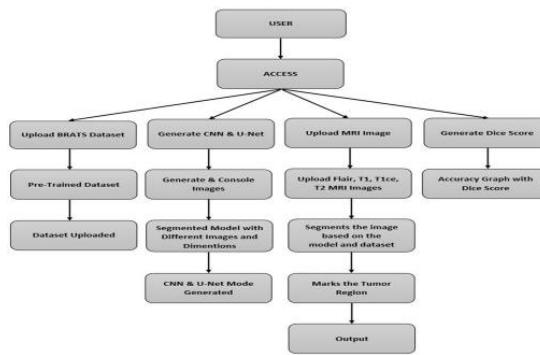


Fig. 1: Proposed System Architecture

A. Upload the BRATS dataset

The dataset is uploaded for creating the system for U-nets using CNN. The dataset contains 4 different kinds of images for each brain scan. These images include the flair, t1, t1ce, t2. All the images are in black-and-white format only. These images are preprocessed and reduced for efficient processing. For this project, a total of 138 photos were shot, which is a substantial number given that each image is further divided into numerous pixels.

B. Generate CNN & U-net

The next stage is to create the CNN and U-net models. In the U-net model, the initial stage is called downsampling, during which the area of the images is decreased. The goal of downsampling is to preserve the image's context; as a result, the feature maps get smaller, making it easier for us to comprehend the image's overall intricacy and structure. A number of ReLU-type activation functions and procedures like pooling (such as Max pooling) are used to accomplish downsampling. Each of these layers continues to apply filters to the images, decreasing their dimensionality and producing feature maps that produce information that is similar but more compressed. The process of upsampling comes after the downsampling. Upsampling is done to make localization easier and more accurate. This is used by networks to produce high-resolution images from compressed feature maps. This is critical for both segmentation and boundary drawing to emphasize the areas that have been divided. A sequence of convolutional layers that raise the spatial dimensionality is used to perform the upsampling. Stated differently, it merely expands the feature maps' dimensions. Additionally, it utilizes the feature maps from the earlier downsampling procedure to guarantee the network of high-resolution images (also known as skip connections) and provides accurate localization.

**C. Upload MRI Image (for Segmentation)**

Once the module is generated, we need to upload again 4 different types of images for the same brain scan. These are flair, t1, t1ce, and t2. And the program will output an image with the window title as "label", this shows the actual segmentation that's performed on the image. Once the module is generated, the image is sent for downsampling where it's reduced, then the segmentation takes place during the upsampling part.

D. Generate Dice Score

The dice score is a tool for evaluating how well the system or model performs while creating accurate segmentations. The first step in determining the Dice similarity score between two samples is to figure out which pixels in each sample are constant for the intersection, or object of interest. Next, tally every pixel recognized as an item in every pattern. The number of consensus pixels divided by the complete number of pixels in the two samples yields the dice score. This is a straightforward illustration of how well two pieces match; the range is 0 (no match) to 1 (very exact).

VI. MODULES**A. Upload BRATS Dataset**

Upload BRATS Dataset is the first module of our project, it is used to upload the BRATS dataset. Click on 'Upload BRATS Dataset' button to upload dataset

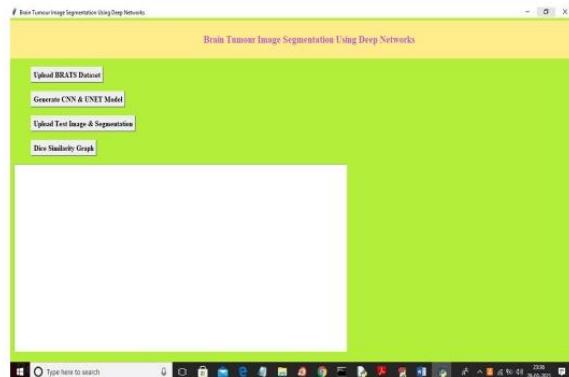


Fig. 2: User Interface (Home Page)

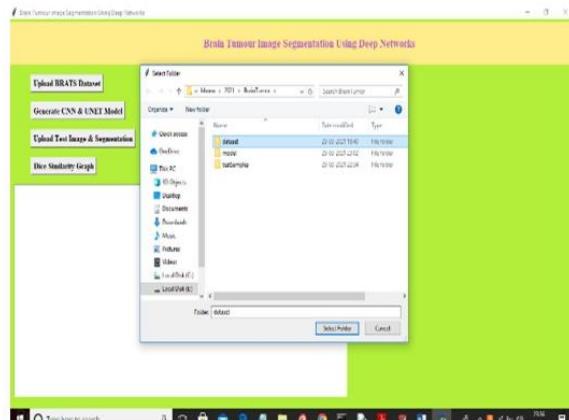


Fig. 3: Upload BRATS Dataset



B. Generate CNN & UNET Model

Generate CNN & UNET Model is the second module of our project, it is used to generate models are generated and console to see CNN and UNET layer details. Click on ‘Generate CNN & UNET Model’ button to generate models, we can see models are using different size images to filter them and to get best features from it to build efficient model and now model is generated.



Fig. 4: Dataset Loaded



Fig. 5: Generate CNN & UNET Model

C. Upload Test Image & Segmentation

Upload Test Image & Segmentation is the third module of our project and then upload test samples to get segmented output. selecting and uploading ‘Sample1’ folder and then click on ‘Select Folder’ button to get below output top 4 images are the input images such as FLAIR, T1, T2 and T1CE and 5th image is the predicted image with segmented part showing in red colour and this algorithm correctly detecting and marking tumour area and now test with other image.

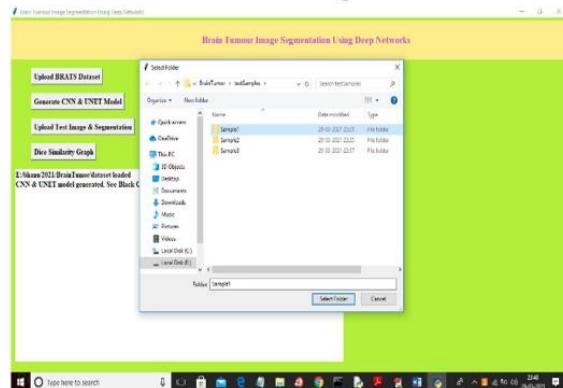


Fig. 6: Upload Test Image & Segmentation

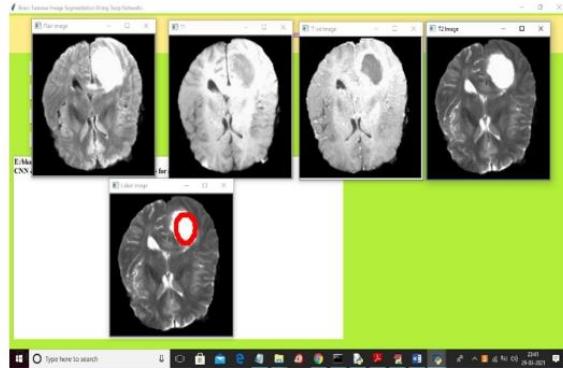


Fig. 7: Result of the Project

D. Dice Similarity Graph

To build CNN and UNET model we took 50 epoch or iterations and at each iteration DICE score between training and testing images get better and better and we get final dice score as $0.8 * 100 = 80\%$. In above graph x-axis represents epoch and y-axis represents dice score.



Fig. 8: Dice Similarity Graph

VII. CONCLUSION

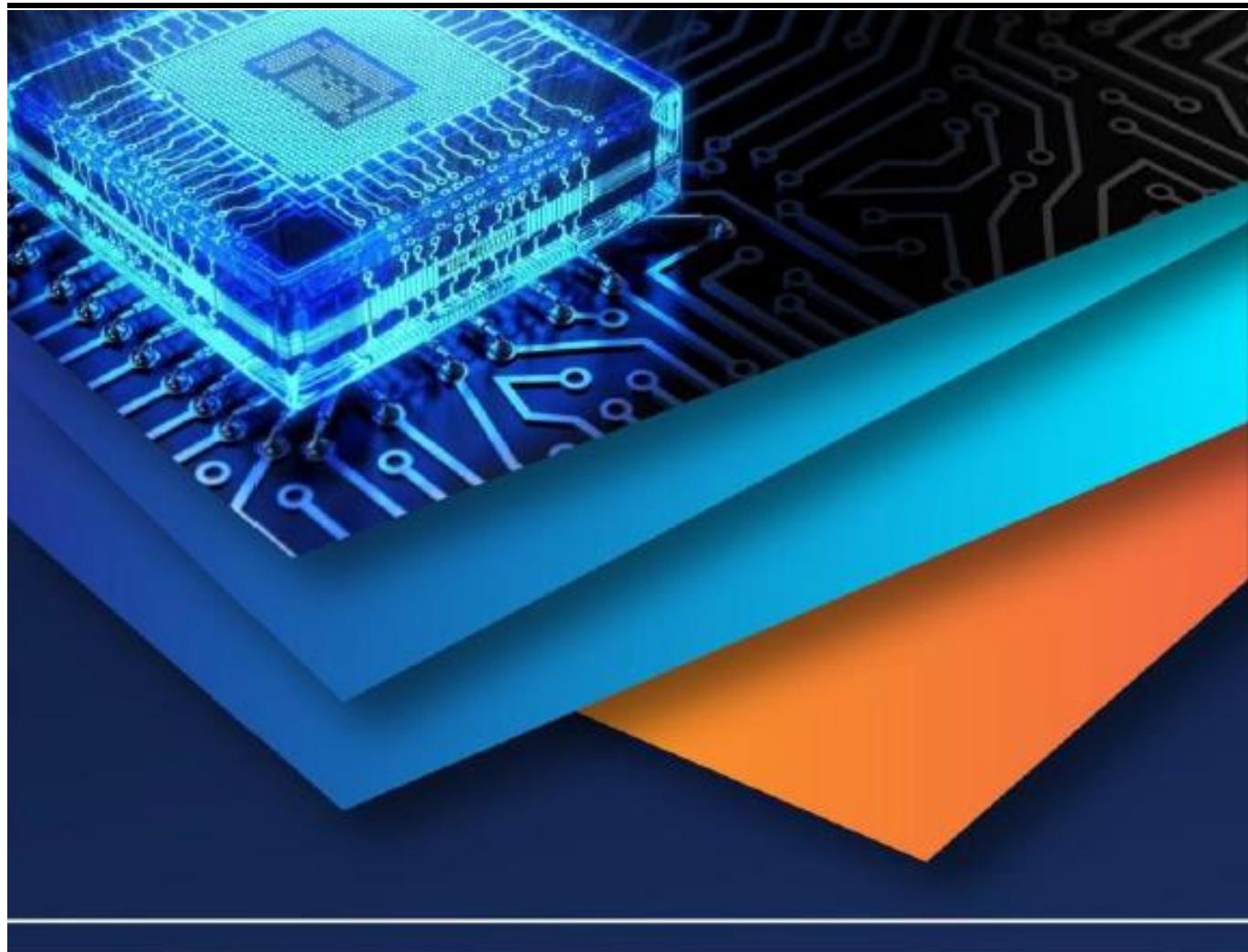
We presented the group of two networks in this study, which are employed together in an ongoing manner to segment biological images. As required by the BraTS 2019 challenge, the group capitally produces a segmentation of brain tumours from the multichannel MRI images that is substantially accurate. This segmentation compares favorably with the predictions provided by other colored state-of-the-art models. To get the style scores, we mix the individual labors from the prototype using a system of variable ensembling. The suggested group provides a clinically useful robotic and objective method for segmenting brain tumors to support complaint planning and case management.

REFERENCES

- [1] "A check of MRI-predicated clinical image study for brain tumour studies," S.Bauer, R.Wiest, L.P Nolte, M.Reyes, 2013, (Online). accessible at <http://www>.
- [2] "Global frequency of nasty brain and other CNS tumours by diagnosis, 2003–2007," Neuro Oncol., vol.19, no.11, pp.1553-1564, 2017, R.Leece, J.Xu, Q.T.Ostrom, Y.Chen, C.Kruchko, and J.S.Barnholtz-Sloan.
- [3] "CBTRUS statistical report first brain and CNS excrescences identify in the US in 2005-2009" (T.A. Dolecek, J.M. Propst, N.E.Stroup, C.Kruchko), Neuro Oncol., vol. 14, no.suppl_5, pp. v1–v49, 2012.
- [4] Acta Neuropathol., vol.131, no.6, pp.803-820, 2016. D.N.Louis et al., "The 2016 World Health Organization type of tumors of the CNS a summary."
- [5] R. Stupp etal, "Adjuvant temozolamide plus attendant radiotherapy for glioblastoma," N Engl J Med., vol.352, no.10, pp.987-996,2005.
- [6] S. Bakasetal, "Comparing the swish machine knowledge algorithms for brain tumour separation, progression evaluation, and complete survival identification in the BRATS challenge,"arXiv Prepr arXiv 1811.02629,2018.
- [7] "A generative prototype for brain tumour separation in multichannel images," by B.H.Menze, K.Van Leemput,D.-A.Weber,N.Ayache, and P.Golland, in International Conference on Clinical Image Computing and Computer base supported Intervention,2010,pp.151-159.



- [8] S. Bakas et al., "Using clever segmentation labels and marconi features to advance the cancer genome atlas glioma glamorous Resonance Imaging collections," *Sci. data*, vol. 4, p. 170117, 2017.
- [9] Cancer Imaging Archives, vol. 286 (2016); S. Bakasetal., "Segmentation labels and marconi advances for thepre-operative reviews of the TCGA-LGG collection."
- [10] S. Bakasetal., "Radiomic features and separation labels for the pre operative reviews of the TCGA-GBM collection." *The Cancer Imaging Archive*, 2017; National Science Data, vol. 4, p. 170117.
- [11] "Semantic Segmentation of Brain Tumor from 3 Dimensional Structural MRI Using U-Net Autoencoder," M. FARZANA, M. J. HOSSAIN ANY, M. T. REZA, and M. Z. PARVEZ, 2020 International Conference on Machine Learning and Cybernetics (ICMLC), Adelaide, Australia, 2020, pp. 137-142, doi: 10.1109/ICMLC51923.2020.9469580.
- [12] The article "Current methods in medical image segmentation" was published in 2000 by D. L. Pham, C. Xu, and J. L. Prince in the Annual Review of Biomedical Engineering.
- [13] The article "Automated medical image segmentation techniques" was published in the Journal of Medical Physics/Association of Medical Physicists of India on March 3, 2010, and was written by N. Sharma and L. M. Aggarwal.
- [14] Zhou, Zhou, Tajbakhsh, Siddiquee, M. M. R., and Liang, J., "Unet++: A nested u-net architecture for medical image segmentation", Deep learning in medical image analysis and multimodal learning for clinical decision support, pp. 3-11, 2018.
- [15] The International Journal of Science and Modern Engineering (IJSIME) published a paper by P. Sapra, R. Singh, and S. Khurana titled "Brain tumor detection using neural network" in 2013.
- [16] Pattern Recognition Letters, 2017, J. Amin, M. Sharif, M. Yasmin, and S. L. Fernandes, "A distinctive approach in brain tumor detection and classification using MRI".
- [17] "A new method for brain tumor identification with MRI images," S. Zhang and G. Xu, *Journal of Biomedical Science and Engineering*, vol.9,no.10, pp.44-52,2016.
- [18] "Brain Tumour Detect and Separation using CNN Architecture and U-Net Architecture," 2023 International Conference on Integrated Intelligence and Communication Systems (ICIICS), Kalaburagi, India, 2023, pp. 1-5, doi: 10.1109/ICIICS59993.2023.10421540, S. R. Totad, Meeradevi, B. J. Sowmya, P. Boppudi, N. Yaswanth, and I. Nivedith.
- [19] "Effective MRI based Brain Tumor Detection using Modified U-Net Model," M. M. Nayak and S. D. K. A., 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon), Mysuru, India, pp. 1-7, doi: 10.1109/MysuruCon55714.2022.9972503.
- [20] "A LeakyReLU based Effective Brain MRI Segmentation using U-NET," M. V. S. Lakshmi, P. L. Saisreeja, L. Chandana, P. Mounika, and P. U, 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2021, pp. 1251-1256, doi: 10.1109/ICOEIS1242.2021.9453079.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)

CERTIFICATES:







