Exploring Deep Learning-based Segmentation Techniques for Brain Structures in MRI Scans

Mohammed Khaled Alshedokhi

[1]Eng.Mohammed Khaled Alshedokhi,, Dept. of Electrical and Computer Engineering, King Abdulaziz University, Jeddah, Saudi Arabia

---***---

Abstract - *This paper delves into the realm of deep learning techniques applied to the segmentation of brain structures in MRI scans, with a specific focus on automating the distinction between healthy and tumor regions. Utilizing the U-Net architecture and various deep learning models, this study investigates the utilization of convolutional neural networks (CNNs) and transfer learning to elevate segmentation accuracy and expedite the process. The Brain Tumor Segmentation (BraTS) dataset serves as the primary resource for training and evaluating the models. The research encompasses data preprocessing, model training involving optimized hyperparameters, and comprehensive evaluation metrics.The outcomes manifest the remarkable superiority of deep learning models, particularly the fine-tuned U-Net, when compared to manual segmentation methods, both in terms of precision and efficiency. The final model exhibits a notable enhancement in the detection of tumor regions, thereby contributing to more prompt and precise diagnostic procedures. This work forms a cornerstone for future advancements in the realm of medical imaging and automated diagnostic tools, with direct implications for enhancing patient care and facilitating treatment planning in the fields of neurology and oncology. This paper not only represents a technological milestone in the realm of medical imaging analysis but also opens avenues for comprehensive tumor characterization, potentially facilitating personalized treatment strategies. It underscores the pivotal role and effectiveness of integrating advanced AI techniques into medical diagnostics, marking a significant stride toward the fusion of technology and healthcare, ultimately leading to improved patient outcomes.*

Key Words: **Computer Vision, Deep Learning, Brain Tumor Segmentation, MRI Scans, Artificial Intelligence.**

1.INTRODUCTION

The brain MRI scan segmentation process enables physicians to identify various parts of the brain of individuals. Brain segmentation is often used to analyze and distinguish healthy and unhealthy parts as well as detect detrimental signs of old age. However, this process requires experts in analyzing and segmenting the brain accurately and may take a long time when a pattern must be established in multiple scans through manual processing. In such cases, computer vision and machine learning techniques can be employed to not only speed up the process but also increase segmentation/detection accuracy. Medical diagnostics has also been aided by such techniques. In this research work, we will analyze various deep learning-based techniques to provide an automated MRI scan segmentation tool that is not only accurate but also feasible for speedy processing. Transfer learning techniques will be explored to quickly compare the performance of the well-established deep learning architectures proposed by the experts for similar tasks. Based upon the findings, a unique or incremental solution will be proposed. Standard medical imaging datasets will be used to analyze the performance. But before starting, the computer vision and segmentation must be defined and how they work using deep learning.

An integral aspect of brain MRI scan segmentation is the precise identification of brain tumors. Brain tumor segmentation plays a crucial role in the medical field, as early and accurate detection of tumors is critical for effective treatment planning and improved patient outcomes. Tumors in the brain can vary greatly in size, shape, and location, making their detection challenging yet vital. Misidentification or late detection of these tumors can lead to severe consequences, including rapid disease progression and reduced treatment options. By focusing on brain tumor segmentation, this research aims to contribute significantly to the early diagnosis and management of brain tumors. Utilizing advanced deep learning techniques in this domain not only promises greater accuracy in segmentation but also ensures a quicker turnaround in diagnostic procedures, thereby potentially improving the survival rates and quality of life for patients afflicted with brain tumors

1.1 Computer Vision and Image Segmentation

Image segmentation is a commonly used technique in digital image processing and analysis to partition an image into multiple parts or regions often based on the characteristics of the pixels in the image [1]. Over the years, image segmentation has been used in many areas such as medical imaging, scene analysis for autonomous vehicles, object detection and tracking, etc. An example of the application of image segmentation in the domain of medical imaging is shown in Fig- 1.
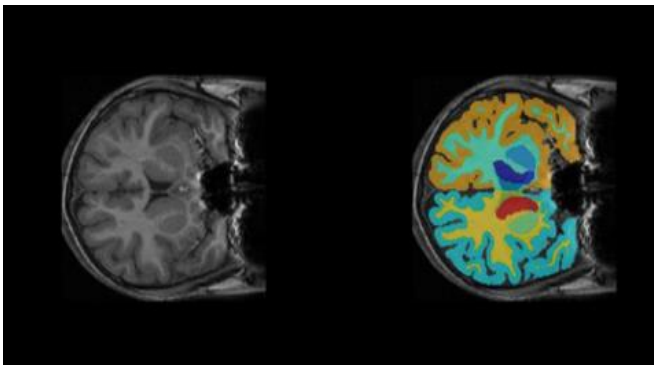
Fig- 2: Medical Imaging

Machine learning is often used for segmentation of images. In recent years, deep learning-based approaches employing specialized neural networks commonly known as convolutional neural networks (CNNs) have outperformed many conventional methods for image segmentation. A CNN is composed of many layers stacked in a particular fashion to achieve a specific task learned with the help of data and ground truth. In the case of segmentation, the output of a CNN is an image where each pixel is assigned a desired class label as shown in Figure 1.2.
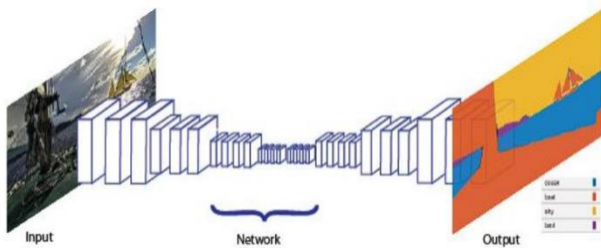

Fig- 2: Deep-learning Segmentation.

Segmentation is of high importance in neuroimaging, and although clinical segmentation is a gold standard in terms of accuracy, it requires experts in neuroanatomy, which means that it requires time, effort, and cost [2]. For these reasons, automated segmentation methods have been proposed as a fast and reproducible alternative solution. As mentioned earlier, modern automation of segmentation is done by using artificial intelligence, specifically using Convolutional Neural Network (CNN) which learns directly from data. Modern automation will enable us to efficiently exploit every brain scan for scientific and research purposes instead of leaving it in the archives [3-4]. With the availability of computing power and storage capacity, we are able to scan and process millions of samples contrary to tens or hundreds in a clinical environment. Therefore, there is a clear need for a fast, accurate, and reproducible automated method for segmentation of brain scans of any contrast and resolution and that can adapt to a wide range of patients [5-6].

Finally, complete content and organizational editing before formatting. Please take note of the following items when proofreading spelling and grammar:

1.2 Problem Statement

The manual examination of MRI brain scans by medical experts, although meticulous, is time-consuming and can lead to extended diagnosis periods. The challenge is amplified when distinguishing between healthy tissue and non-healthy (tumor) parts, where precision is critical for effective diagnosis and treatment planning. Furthermore, manual analysis limits the ability to rapidly process large volumes of scans, which is essential in research and clinical settings. Therefore, this research emphasizes the need for an automated brain MRI scan segmentation tool. Such a tool aims to efficiently differentiate between healthy and non-healthy (tumor) regions in the brain. By leveraging deep learning techniques, this project seeks to develop a solution that not only accelerates the segmentation process but also maintains, or potentially improves, the accuracy compared to traditional manual methods. This approach aims to expedite diagnosis and enhance the capacity for handling a larger number of cases, ultimately contributing to better patient outcomes in the field of neurology and oncology.

2. RESEARCH METHODOLGY

In this chapter will cover BraTS dataset used in the experiment, and how the processing is done to get the data for training stage, and a definition of what the UNet network is and what it serves. In conclusion, we talked about the testing mechanism that was done for this structure.

2.1 Brain MRI Dataset

The BraTS (Brain Tumor Segmentation) dataset stands as a cornerstone in the realm of medical image analysis, specifically tailored for the challenging task of brain tumor segmentation within magnetic resonance imaging (MRI) scans. Developed to foster advancements in algorithmic approaches for automated and semi-automated segmentation, BraTS has become a pivotal benchmark dataset in the field.[10] There are different modalities of medical scan, such as Computed Tomography (CT), Positron Emission Tomography(PET) and Magnetic Resonance imaging (MRI). The BraTS uses the MRI due to the clarity of tissues and tumors in the medical images.[9] As shown in the Fig-3 below.
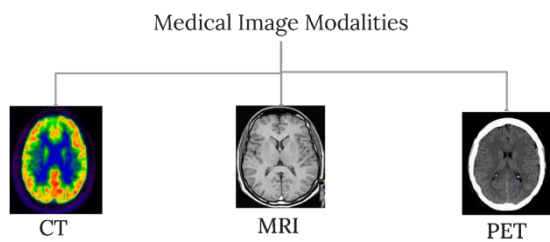
Fig- 3: Medical image Modalities

Researchers and developers can access the BraTS dataset for research and development purposes through the Center for Biomedical Image Computing and Analytics (CBICA) at the University of Pennsylvania[10]. This combination of resources offers a comprehensive platform for global researchers, facilitating significant advancements in medical imaging and diagnostic methodologies.

2.1.1 MRI Scan

Brain MRI primarily relies on a magnetic field, in contrast to CT scans and other methods that utilize radiation. It is effective in identifying brain tissue swelling, infections, and tumors, making the resulting MRI images crucial for analyzing various types of brain abnormalities. In the brain MRI procedure, large magnets generate a magnetic field that ranges from 0.2 Tesla (T) to 7 Tesla (average 1.5 Tesla). The term 'Tesla' here refers to the unit of measurement for magnetic field strength. One Tesla represents a very strong magnetic field. The subject undergoing the MRI is placed inside this strong magnetic field, and the excited hydrogen atoms in the body – due to the high water content – emit radio frequencies. These frequencies are captured within the large enclosed area of the MRI scanner, allowing for detailed imaging. The procedure to get an MRI scan is depicted in Fig-4
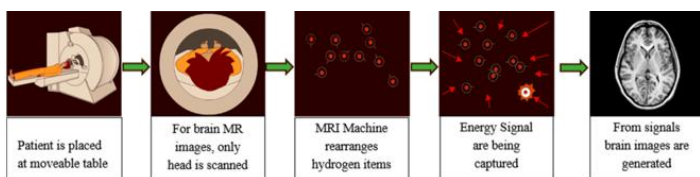

Fig- 4: MRI Process

While taking an MRI scan of the patient, the machine will output four different medical images depending on the frequency and time , namely: T1,T2,T2c and Flair [9].
- o **T1**: This type has a small echo and repetition time (Nice image contrast) [9]
- o **T2**: It has a long time of echo and repetition time but slow image acquisition.[9]
- o **T1c**: Same as T1, but a contrast agent is applied to enhance the contrast.
- o **Flair**: Used to nullify the signal from the fluid.

2.1.2 BraTS 2017 dataset

The BraTS is collected from multiple institutions, providing a diverse set of cases and variations. Dataset contains 4 classes, from which the types of tumors are classified as Necrosis, Edema, Non-enhancing and enhancing.[11]
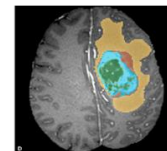

Fig- 5: Glioma Tumor Types

Each MRI scan in the BraTS dataset is manually annotated by expert radiologists to provide ground truth segmentation masks for different tumor sub-regions. These sub-regions typically include the necrotic core, edema, enhancing tumor, and the non-enhancing tumor.[11] The dataset contains 750 4D cases, and its size is **240x240x155x4** which means that the number of images for each case is **620**, detailed below: -

- o **240x240:** Single image size.
- o **155**: Number of slices.
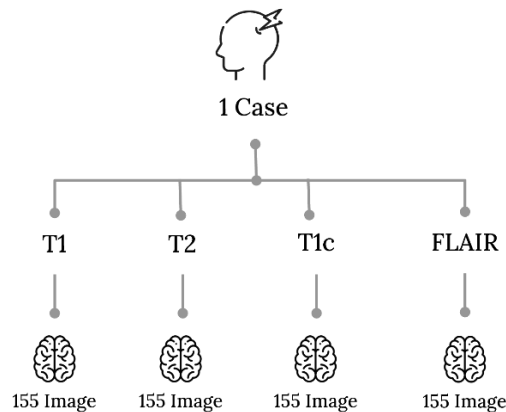- o **4:** type of image {T1, T1c, T2 and Flair}.


Fig- 6: Single Case Details- Brats

The BraTS dataset is divided into 484 labeled cases for training (80%) and 289 cases for testing (20%). As shown in the summary table below: -

Table -1: BraTS2017 Dataset summary

BraTS2017 Dataset Summary	
Number of Total Cases	750
Number of **Training** Cases	484(80%)
Number of **Testing** Cases	289(20%)
Total Number of MR Images	465,000
Modalities for Each Case	4 {T1, T1c, T2, FLAIR}
MR Pixel Resolution	240x240
MR Image in each Modality	155
Total MR for Each Case	620

2.2 Preprocessing

The pre-processing stage is a crucial component in the development of robust algorithms for brain tumor segmentation using the BraTS (Brain Tumor Segmentation) dataset. Pre-processing tasks aim to enhance the quality and consistency of the data, address variations in imaging characteristics, and prepare the dataset for effective training and evaluation. The following are key pre-processing steps commonly applied to the BraTS dataset. During the preprocessing process, two important stages pass: preparing data, then applying Channel Wise PreProcess, as shown in the Fig-7 below.
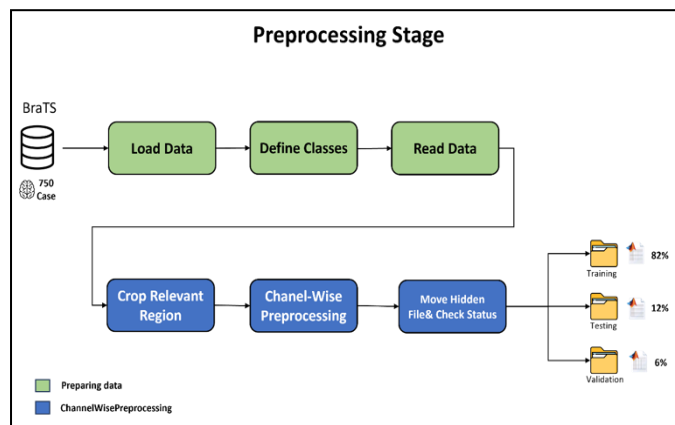
Fig- 7: Preprocessing Chart

2.2.1 Preparing Data

In the data preparation phase for the BraTS (Brain Tumor Segmentation) dataset, the process begins with the organization and preparation of MRI scan images and their corresponding labels. The initial step involves identifying and arranging the volumetric images and labels into appropriate categories. This organization is crucial to ensure that each image is correctly matched with its corresponding label, which is essential for accurate segmentation and analysis.

The next key step is the refinement of these images. This involves focusing on the relevant areas of the brain and the tumor by eliminating unnecessary background elements. Such refinement is vital as it helps in isolating the brain structures and tumors, making them more prominent for the subsequent analysis. This step is particularly important as it enhances the clarity and focus of the images, ensuring that the analysis is concentrated on the most relevant parts of the scans. Finally, the prepared data is systematically arranged into training, test, and validation sets. This segregation is fundamental for the effective training and evaluation of any segmentation models developed using this dataset. The training set is used to train the model, the validation set helps in tuning the model parameters, and the test set is crucial for evaluating the model's performance. Overall, this meticulous approach to data preparation is essential for ensuring the quality and reliability of the dataset. It sets a strong foundation for the further steps in the research process, enabling more accurate and effective analysis of brain tumors using the BraTS dataset.

2.2.1 ChannelWisePreProcess.

In the next phase of data preparation, the MRI scans from the BraTS dataset undergo a crucial normalization process. This process is applied channel-wise to each modality of the MRI scans. Normalization is an essential step in image processing, especially in medical imaging, as it helps to standardize the data, making it more uniform and easier to analyze.

The normalization procedure involves two main steps. First, for each channel of the MRI scan, the mean value of the pixel intensities within the cropped brain region is calculated and subtracted from each pixel. This step centers the data around zero. Following this, each pixel value is divided by the standard deviation of the pixel intensities in the cropped area. This scaling ensures that the data has a consistent variance, which is critical for many analytical methods.

To further refine the data, any outliers in pixel values are adjusted to fall within a pre-defined range. This clamping of values helps in mitigating the effects of extreme values which can skew analysis. Once these steps are completed, the pixel values are rescaled to fit within a standardized range, typically between 0 and 1. This rescaling is beneficial for various deep learning models which are often sensitive to the scale of input data.

After normalization, the dataset is divided into three parts: training, validation, and testing sets. This division is typically done in a manner where a majority of the data,

around 82%, is used for training the models, about 6% for validating the model's performance during the training process, and the remaining 12% for testing the model's effectiveness on unseen data.

Finally, the processed and normalized data is saved in a format suitable for further analysis and model training. This step ensures that the data is readily available in a structured form for the subsequent stages of the research.

2.3 UNet Network [12]

Convolutional networks are often used to classify images by giving them one label that represents the whole image. But in tasks like biomedical image processing, it's important to know the label for each tiny part of the image (Localization), called pixels. Getting lots of training images for biomedical tasks is usually hard. So, Ciresan and others came up with a way. They trained a network to figure out the label for each pixel by looking at a small area (patch) around that pixel in a sliding-window setup. [12]
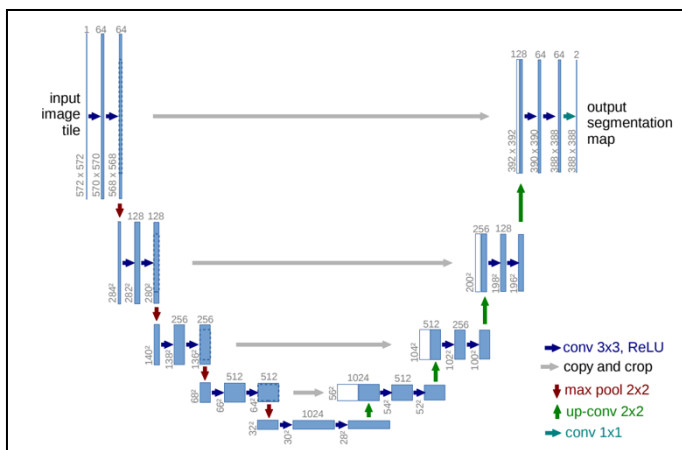

Fig- 8: UNet Architecture

Training the U-Net architecture for biomedical image segmentation tasks is a detailed process that involves teaching the network to classify each pixel of an image. This pixel-level classification is crucial for accurately identifying and segmenting regions of interest, such as tumors or organs, in medical images. The training process typically involves the following steps:

Data Preparation: The network is trained on a set of labeled images, where each pixel is categorized, for instance, as part of a tumor or healthy tissue. These labels serve as the ground truth for training the network.

Patch-Based Training: Given the challenge of obtaining a large number of biomedical training images, U-Net often employs a patch-based approach. This method involves training on small regions or patches extracted

around each pixel, which helps the network learn from a limited dataset by focusing on local features and contexts.

Loss Function: The choice of a loss function is critical in training U-Net. For biomedical image segmentation, the Dice Coefficient Loss is commonly used. This loss function is particularly suited for dealing with the class imbalance often found in medical images, where the region of interest (like a tumor) occupies a much smaller portion of the image compared to the background. The Dice Coefficient Loss is formulated to maximize the overlap between the predicted segmentation and the actual ground truth, making it a robust choice for this task.

Network Training: The training involves feeding the network with batches of these patches and adjusting the network weights based on the loss calculated from the predictions. The network learns to classify each pixel accurately through this iterative process of prediction and adjustment using backpropagation.

Iterative Optimization: This training is conducted over multiple iterations, known as epochs, where the network continually improves its ability to segment the images accurately.

U-Net's design, with its contracting and expansive paths and skip connections, enables it to capture both the context and fine details required for precise pixel-level segmentation. The model's ability to be trained effectively on relatively smaller datasets while producing accurate segmentation results has made it a preferred choice in various biomedical image segmentation applications.

2.3.1 Contracting Path (Encoder)

This part of the network consists of convolutional and pooling layers, which reduce the spatial resolution while increasing the number of channels. This helps in extracting features and capturing context.[12]

2.3.2 Expansive Path (Decoder)

The expansive path involves upsampling and concatenation operations, which gradually increase the spatial resolution while reducing the number of channels. This helps in precise localization.[12]

2.3.3 Bottleneck

The bottleneck connects the contracting and expansive paths and acts as a bridge between the two. It is often a simple convolutional layer.[12]

2.3.4 Skip Connections.

Skip connections, also known as residual connections, connect the corresponding layers in the contracting and expansive paths. These connections enable the network to retain high-resolution information during the upsampling process.[12]

U-Net is particularly popular in biomedical image segmentation tasks, such as segmenting organs or tumors in medical images. Its ability to handle relatively small datasets and produce accurate segmentation results has contributed to its widespread adoption in various applications beyond medical imaging. [12]

2.4 Evaluation.

Evaluation is a critical aspect of assessing the performance of a semantic segmentation model. In this section, we delve into the metrics used to quantify the accuracy of the UNet network in segmenting brain tumor images. The evaluation metrics include Global Accuracy, Mean Accuracy, Mean Intersection over Union (Mean IoU), and Weighted Intersection over Union (Weighted IoU). These metrics offer a comprehensive understanding of the model's ability to classify pixels into different classes, considering both overall and class-specific performance.[13]

2.4.1 Global Accuracy

Global Accuracy is the percentage of correctly classified pixels over the entire image dataset. Global Accuracy is calculated in the following formula: -[13]

$$Globa\ Accuracy = \frac{Number\ of\ Correctly\ Classified\ Pixels}{Total\ Number\ of\ Pixels} \times 100\%$$

Global Accuracy provides an overall measure of how well the model is performing, but it might not be sufficient in the presence of class imbalance.[13]

2.4.2 Mean Accuracy

Mean Accuracy is the average accuracy for each class and then averages these values.[13]

$$Mean\ Accuracy = \frac{1}{N} \sum_{i=1}^{N} \frac{True\ Positives_i}{True\ Positives_i + False\ Positives_i}$$

Where N is the number of classes = 2

Mean Accuracy gives a more nuanced view of accuracy, considering the accuracy for each class individually.[13]

2.4.3 Mean Intersection over Union

Mean Intersection over Union (mIoU) is computed by first calculating the Intersection over Union (IoU) for each semantic class. Intersection over Union (IoU), also known as the Jaccard index, is the most popular evaluation metric for tasks such as segmentation, object detection and tracking. Object detection consists of two sub-tasks: localization, which is determining the location of an object in an image, and classification, which is assigning a class to that object.[14]

Mean IoU measures the average of the Intersection over Union (IoU) for each class.[5]

$$Mean\ IoU = \frac{1}{N} \sum_{i=1}^{N} \frac{True\ Positives_i}{True\ Positives_i + False\ Positives_i + False\ Negatives_i}$$

Where N is the number of classes = 2

Mean IoU considers both false positives and false negatives, providing a more comprehensive measure of segmentation accuracy. [13]

2.4.4 Weighted intersection over Union

Weighted Intersection over Union (Weighted IoU):
Weighted IoU is an extension of Mean IoU that considers class imbalance by assigning different weights to different classes.[13]

$$Weighted\ IoU = \frac{1}{\sum_{i=1}^{N} Weight_i} Weight_i \times \frac{True\ Positives_i}{True\ Positives_i + False\ Positives_i + False\ Negatives_i}$$

Where N is the number of classes = 2

Weighted IoU useful when there is a significant imbalance in the number of pixels belonging to different classes.[13]
These metrics collectively provide a detailed evaluation of the model's segmentation performance, considering both overall accuracy and class-specific performance. The Weighted IoU is particularly valuable when there are imbalances in the distribution of different classes.[13]

3. RESULT AND DISCUSSION

This chapter will cover the experiments and discuss the results. As is known, research experiments related to artificial intelligence depend on changing the hyperparameter after designing the methodology and recording the results. This iterative process is aimed at enhancing accuracy and evaluating the efficacy of the model structure. We have four metrics to evaluate the success of the model, as mentioned in the previous chapter, Global Accueracy, Mean Accuracy, Mean IoU and Weighted IoU. Before delving into the specifics of the experiments conducted, it's crucial to understand the general mechanism of training in neural networks, particularly in the context of image segmentation tasks.

Training Mechanism Overview: Neural networks, like the U-Net used in our experiments, are designed to learn the correspondence between input images and their corresponding labels. In the case of image segmentation, the input is the image, and the label is the segmentation map of the image, which indicates the category of each pixel (e.g., tumor or non-tumor). The network learns this correspondence through a process known as backpropagation.

Backpropagation Process: During training, the network makes predictions based on its current state (the weights of the neurons). These predictions are compared against the actual labels, and the difference (error) is calculated. Backpropagation is then used to propagate this error back through the network, allowing it to adjust its weights in a way that reduces the error in subsequent predictions. This process is repeated iteratively over many cycles, known as epochs, gradually improving the network's ability to accurately predict the labels from the images.

Role of Hyperparameters: In addition to the weights of the network, which are learned during training, there are hyperparameters that need to be set before the training begins. Hyperparameters are not learned from the data but are set by the experimenter and have a significant impact on the performance of the network. These include the patch size, mini-batch size, encoder depth, number of epochs, learning rate, etc. Selecting the right set of hyperparameters is crucial, as they govern various aspects of the training process and the network's architecture.

Hyperparameter Tuning Through Experimentation: The optimal values for these hyperparameters are not known in advance and are usually found through experimentation.

This involves training the network multiple times with different sets of hyperparameters, evaluating its performance each time, and then selecting the set of hyperparameters that yields the best results.

The training of a neural network for image segmentation is a complex process that involves learning the weights of the network through backpropagation and carefully selecting the hyperparameters that guide this training. This process is essential for the network to effectively learn the correspondence between images and their segmentation labels. After choosing the UNet and BraTS Dataset, now we must change the hyperparameter and verify the results, so the experiment passes through these stages: -

1. Determine Hyperparameters
2. Determine number of Epochs and Learning Rate
3. Training the Model with 484 cases
4. Testing the Model with 155 cases
5. Validation Procedures

6. Results Documentation and the initiation of a new experimental cycle by returning to step 1.

The hyperparameters that we will change during each experiment are:

• **Patch Size**: Refers to the dimensions (width and height) of a small, rectangular subset of an image.

• **Patch per Image**: Indicates the number of non-overlapping or overlapping patches extracted from a single image.

• **Mini-Batch Size:** The number of examples (data points or samples) used in one iteration of training a neural network.

• **Encoder Depth**: The number of layers in the encoder part of a neural network architecture

• **Epoch**: Single pass through the entire training dataset during the training of a machine learning model

• **Learning Rate**: Determines the size of the steps taken during the optimization process of training model

The testing phase included a dynamic showcase of the model's real-time predictions alongside the actual outcomes. Numerous experiments were conducted in pursuit of optimal results, and in this chapter, we meticulously examine the most noteworthy findings derived from these endeavors.

3.1 Experiment1 and Discussion (Default Configuration)

In this experiment, the default parameters were set as the starting point, and we started improving.

Table -2: Ex1 Hyperparameters

Hyperparameter	Value
Patch size	[128 128]
Patch per image	2
Mini batch size	100
Encoder depth	3
Number of epoch	5
Learning rate	0.0001

Results of Evaluation: -

Table -3: Ex1 Result

Global Accuracy	Mean Accuracy	Mean IoU	Weighted IoU
99.6%	95.6%	87.2%	99.2%

The default configuration showcases impressive performance, with both high global accuracy and Weighted IoU. It indicates that the model is well-tuned with the initially chosen hyperparameters, providing a solid baseline. But we noticed that Mean IoU did not give the high result that the rest of the evaluations gave. Mean IoU is a metric that measures the overlap between the predicted segmentation and the ground truth, providing an indication of how well the model is delineating the boundaries of the segmented regions.

3.2 Experiment2 and Discussion (Changing patch size)

In this experiment, I changed the patch size, and analyzed the results. I experimented with increasing the patch size, then decreasing it while keeping the other hyperparameters constant.

Table -4: Ex2 Hyperparameters

Hyperparameter	Value
Patch size	[176 176]
Patch per image	2
Mini batch size	100
Encoder depth	3
Number of epoch	5
Learning rate	0.0001

Results of Evaluation: -

Table -5: Ex2 Result

Global Accuracy	Mean Accuracy	Mean IoU	Weighted IoU
99.8%	94.5%	87.4%	99.6%

While changing the patch size, I found that the best way to achieve our goal, which is to increase accuracy, is to increase the patch size. I increased it to 176x176 because there is a condition that the patch size must be a multiple of 2^(Encoder Depth). I noticed that all results improved slightly, but the Mean Accuracy declined. But the conclusion in this experiment is that increasing the patch size will be suitable for our goal, unlike decreasing it.

3.3 Experiment3 and Discussion (Changing patch per image)

In this experiment, I increased the Patch per Image, and analyzed the results. I experimented with increasing the Patch per Image, then decreasing it while keeping the other hyperparameters constant.

Table -6: Ex3 Hyperparameters

Hyperparameter	Value
Patch size	[176 176]
Patch per image	4
Mini batch size	100
Encoder depth	3
Number of epoch	5
Learning rate	0.0001

Results of Evaluation: -

Table -7 Ex3 Result

Global Accuracy	Mean Accuracy	Mean IoU	Weighted IoU
99.8%	93.6%	87.7%	99.6%

After increasing the number of patches per image, no noticeable improvement was observed in the four evaluations. On the contrary, the Mean Accuracy decreased by approximately 1%. From this experiment, we concluded that the previous number of patches for each image, which is 2, is appropriate.

3.4 Experiment4 and Discussion (Maximize Encoder Depth & Patch Size)

After the success of experiment No. 2, which is increasing the patch size, in this experiment we will set the largest value for the Encoder Depth and Patch size. We have a limit, which is the original image size, which is [240 240], and the patch size restrictions are: -

1- It must be a multiple of 2^(Encoder Depth).
2- It must not exceed the size of the original image
Based on these restrictions, the maximum value we can set for the Encoder Depth is 6
The only value we are allowed to set for the patch size is [192 192]

Table -8: Ex4 Hyperparameters

Hyperparameter	Value
Patch size	[192 192]
Patch per image	2
Mini batch size	100
Encoder depth	6
Number of epoch	5
Learning rate	0.0001

Results of Evaluation: -

Table -9 Ex4 Result

Global Accuracy	Mean Accuracy	Mean IoU	Weighted IoU
99.85%	95.1%	89.9%	99.7%

The results from this experiment were highly encouraging, showing significant improvements across all evaluation metrics, including Global Accuracy, Mean Accuracy, Mean IoU, and Weighted IoU. Notably, there was a marked increase in the Mean IoU, a metric that had not shown as much improvement in previous experiments. This suggests that increasing both the encoder depth and the patch size positively impacts the model's accuracy. Therefore, the experiment concludes that maximizing these hyperparameters is effective in enhancing the model's segmentation accuracy.

3.5 Experiment5 and Discussion (Increasing number of epoch)

After the amazing result in the fourth experiment, increasing the number of Epochs will lead to an increase in training time, but will improve the results. In this experiment, I increased the number of Epochs to 50.

Table -10: Ex5 Hyperparameters

Hyperparameter	Value
Patch size	[192 192]
Patch per image	2
Mini batch size	100
Encoder depth	6
Number of epoch	50
Learning rate	0.0001

Results of Evaluation: -

Table -11 Ex5 Result

Global Accuracy	Mean Accuracy	Mean IoU	Weighted IoU
99.9%	96%	91.3%	99.8%

Even after increasing the number of Epochs, the improvement was amazing compared to the fourth experiment. Of course, training the model took 10 times the time it took in the fourth experiment, but the improvements we obtained were worth it, and the table below shows the differences between the default model and the improved model (the fifth experiment).

Table -12: Models compression

Evaluation	Default Model (Ex1)	Improved Model (Ex5)	Enhancement %
Global Accuracy	99.6%	99.9%	0.3%
Mean Accuracy	95.6%	96%	0.4%
Mean IoU	87.2%	91.3%	4.7%
Weighted IoU	99.3%	99.8%	0.5%

As shown in the table above, there was a noticeable improvement in the Mean IoU, it improved by 4%, and the other evaluations all improved slightly. Of course, as the Epoch increases, the training time increases, but training the model takes place only once. In medical applications the accuracy factor is a very important factor compared to the training time.

4. CONCLUSIONS

This research marks a significant stride in leveraging technology to enhance global health standards. Through the development of a robust automated tool for segmenting brain MRI scans using the U-Net architecture, we have demonstrated how advanced deep learning techniques can be effectively applied in medical diagnostics. The U-Net network, fine-tuned with specific hyperparameters, exhibited substantial accuracy gains in differentiating between tumor and non-tumor regions in brain images.

The quick and accurate diagnosis of tumors, as facilitated by this research, is a crucial step towards timely and effective treatment. Automated diagnosis tools like the one developed here play a pivotal role in healthcare, offering advantages in speed, accuracy, cost-effectiveness, and widespread availability. As medical imaging technology continues to evolve, such automation will be integral to ensuring prompt and precise diagnoses, ultimately enhancing patient care and advancing medical research.
Looking forward, there is potential for deeper exploration in this field. Future work could use the same dataset for more detailed segmentation and localization of specific types of brain tumors, such as Necrosis, Edema, Enhancing, and Non-enhancing tumors. This advancement would not only refine the segmentation process but also aid in providing more detailed insights for targeted medical treatments.

In conclusion, this research underscores the transformative impact of integrating technology in medical fields. It highlights how tools like the U-Net network can revolutionize the diagnosis and treatment process, ultimately raising the standard of global health.

REFERENCES

[1] Image Segmentation. (n.d.). MATLAB & Simulink. https://www.mathworks.com/discovery/image-segmentation.html

[2] Billot, B., Greve, D. N., Puonti, O., Thielscher, A., Koen, V. L., Fischl, B., Dalca, A., V., & Iglesias, J. E. (2021). SynthSeg: Domain Randomisation for Segmentation of Brain Scans of any Contrast and Resolution. Cornell University - ArXiv. https://doi.org/10.48550/arxiv.2107.09559

[3] M. Ghaffari, A. Sowmya and R. Oliver, "Automated Brain Tumor Segmentation Using Multimodal Brain Scans: A Survey Based on Models Submitted to the BraTS 2012–2018 Challenges," in IEEE Reviews in Biomedical Engineering, vol. 13, pp. 156-168, 2020, doi: 10.1109/RBME.2019.2946868.

[4] N. Yamanakkanavar and B. Lee, "Using a Patch-Wise M-Net Convolutional Neural Network for Tissue Segmentation in Brain MRI Images," in IEEE Access, vol. 8, pp. 120946-120958, 2020, doi: 10.1109/ACCESS.2020.3006317.

[5] F. Ramzan, M. U. G. Khan, S. Iqbal, T. Saba and A. Rehman, "Volumetric Segmentation of Brain Regions From MRI Scans Using 3D Convolutional Neural Networks," in IEEE Access, vol. 8, pp. 103697-103709, 2020, doi: 10.1109/ACCESS.2020.2998901.

[6] A. Huang, R. Abugharbieh and R. Tam, "A Hybrid Geometric–Statistical Deformable Model for Automated 3-D Segmentation in Brain MRI," in IEEE Transactions on Biomedical Engineering, vol. 56, no. 7, pp. 1838-1848, July 2009, doi: 10.1109/TBME.2009.2017509.

[7] Y. Liu, F. Mu, Y. Shi and X. Chen, "SF-Net: A Multi-Task Model for Brain Tumor Segmentation in Multimodal MRI via Image Fusion," in IEEE Signal Processing Letters, vol. 29, pp. 1799- 1803, 2022, doi: 10.1109/LSP.2022.3198594.

[8] V. Cherukuri, P. Ssenyonga, B. C. Warf, A. V. Kulkarni, V. Monga and S. J. Schiff, "Learning Based Segmentation of CT Brain Images: Application to Postoperative Hydrocephalic Scans," in IEEE Transactions on Biomedical Engineering, vol. 65, no. 8, pp. 1871-1884, Aug. 2018, doi: 10.1109/TBME.2017.2783305.

[9] G. Latif, "DeepTumor: Framework for Brain MR Image Classification, Segmentation and Tumor Detection," Diagnostics, vol. 12, no. 11, p. 2888, Nov. 2022, doi: https://doi.org/10.3390/diagnostics12112888.

[10] www.med.upenn.edu. (n.d.). MICCAI BraTS 2017: Data | Section for Biomedical Image Analysis (SBIA) | Perelman School of Medicine at the University of Pennsylvania. [online] Available at: https://www.med.upenn.edu/sbia/brats2017/data.html [Accessed 10 Nov. 2023].

[11] B. H. Menze et al., "The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS)," IEEE Transactions on Medical Imaging, vol. 34, no. 10, pp. 1993–2024, Oct. 2015, doi: https://doi.org/10.1109/tmi.2014.2377694.

[12] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," arXiv.org, May 18, 2015. https://arxiv.org/abs/1505.04597

[13] "Evaluate Semantic Segmentation Results," Mathworks.com, 2013. https://www.mathworks.com/help/vision/ref/evaluatesemanticsegmentation.html

[14] giou.stanford.edu. (n.d.). Generalized Intersection over Union. [online] Available at: https://giou.stanford.edu/.