An Improvement of The MRI Segmentation Algorithm on The Human Brain

Khawla Hussein Ali[1], Assalah Zaki Atiyah[2]

[1]Asst. Prof. Department of Computer Science, The College of Pure Science, University of Basrah.

[2]M.S. Student, Computer Science Department, The College of Pure Science, University of Basrah.

Abstract - The complexity of segmenting a brain tumor is critical in medical image processing. Treatment options and patient survival rates can only be improved if brain tumors can be prevented and treated. Segmentation of the brain is the most complex and time-consuming task to diagnose cancer utilizing a manual approach for numerous Magnetic Resonance Images. Segmentation of brain tumors must be done automatically. Brain tumor analysis was done using with state-of-the-art Artificial Intelligence algorithm like U-Net with EfficientNet-B7. Brain lesion segmentation is performed using the BraTS 2020 dataset. Based on MRI scans of the brain, the tumor segmentation technique is assessed using F_1 score, Dice Loss, and Intersection Over Union (IoU) score. The U-Net with EfficientNet-B7 architecture was shown to be crucial in the treatment of brain tumors, according to the findings of the experiments.

Index Terms - Brain tumor, BraTS, MRI, Segmentation,   U-NET.

1.   Introduction

Brain tumors are abnormal masses of brain tissue. A very hard skull protects the brain. Any development in such a tiny area might cause issues [1]. Four conventional Magnetic Resonance Imaging (MRI) modalities are employed in brain image studies: native T1-weighted (T1), T2 fluid-attenuated inversion recovery (T2-FLAIR), T2-weighted (T2), and post-contrast T1-weighted (T1ce). FIGURE 1 depicts the four different MRI modalities [2]. Automatic methods for segmenting brain lesions are frequently used hand-crafted features such as edges, corners, histogram of gradient, local binary pattern. The classifier is given these features after they have been extracted. The training procedure of the classifier is not affected by the nature of those features. Semantic segmentation is commonly used in medical imaging to identify the precise location and the form of the body's structures and is essential to the proper assessment of medical disorders and their treatment.
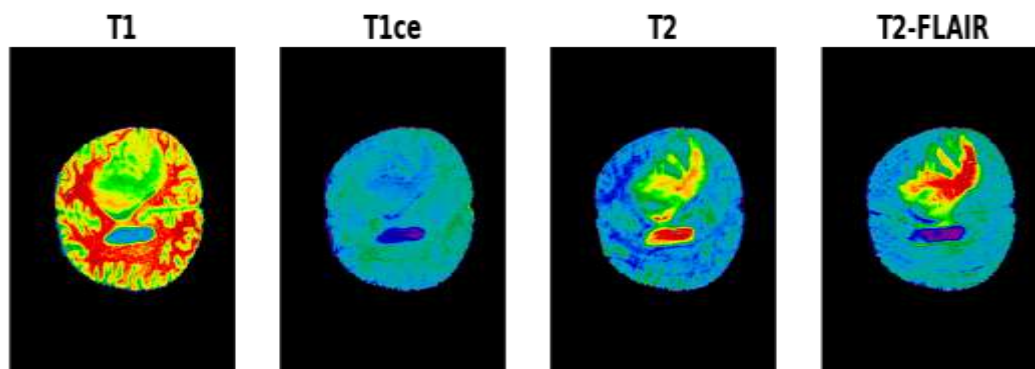

FIGURE. 1 DIFFERENT MRI MODALITIES

Deep CNN has recently achieved substantial advancements in a range of computer vision applications, particularly object detection, image classification, and semantic segmentation, to name a few examples. With their remarkable performance and high accuracy in image segmentation, deep neural networks have recently gained popularity among researchers. CNN refers to a deep neural network, and it is capable of learning to recognize and differentiate characteristics in images. All previous research on this dataset has focused on tumor type detection and classification. Very few of the prior research executed on this dataset is intended for segmentation purposes.

Detection and segmentation of the human brain is a difficult problem to solve in reality due to its complexity. When it comes to tumor-bearing data, it is the 3D or 2D information that varies widely from patient to patient in terms of the structure, size, and location of the tumor. In addition, MRI information of brain tumors extracted from diagnostic scans or synthetic databases is inherently complex, requiring a large amount of device memory for tumor segmentation. We provided U-Net with additional EfficientNet-B7 for both HGG and LGG images in this study. In the proposed method, we augmented the brain MRI images, performed data pre-processing methods to modify the actual data, evaluated different deep learning architectures, and offered a comparative study of these models.

This paper was conducted with various architectures for brain tumor segmentation. Enhanced output was achieved by the proposed architecture. It takes just a few seconds to segment the image using a trained model. Clinical professionals can take hours for manual segmentation of tumors, however. In the field of image diagnosis, this article is contributing to a model that can more accurately and efficiently diagnose the tumor.

This paper discusses techniques for segmenting brain tumors. This article is broken down into many parts. The introduction and background of brain tumors are described in section 1, the literature review is described in section 2, followed by a description of the proposed systems in section 3, with an explanation of the architectures, datasets, training and implementation details, and performance metrics, the experimental results are described in section 4, and the conclusion and future work are described in section 5.

2. Literature Review

Over the last few decades, research into the automatic segmentation of brain tumors has increased, showing an increasing demand for this field of study, which is still ongoing. Several strategies for detecting and segmenting tumors on MRI Image data have indeed been proposed in the research. The majority of brain tumor segmentation methods rely on handcrafted characteristics, that are entered into a classifier, such as the random forest (RF). RF has the best segmentation performance of any traditional classifier.

Brain tumor segmentation has been addressed in the past using several deep learning models, with varying degrees of success.

For the BraTS Challenge 2020 segmentation problem, Fabian et al. [3] utilized nnU-Net in 2021. The nnU-Net pipeline's segmentation performance has been demonstrated to be greatly enhanced by the addition of BraTS-specific characteristics such as postprocessing, data augmentation, and region-based training. Excellent results have been obtained using the nnU-Net configuration's baseline setup.

In 2018, for brain tumor segmentation in MR images, Le et al. [4] devised a learning-based automatic technique. The features extracted from U-Net-based deep convolutional networks were applied to the Extremely randomized trees (Extra Trees) classifier as input data in this method. We also used simple morphological filters to refine the segmentation findings by deleting incorrect labels. The model achieved average Dice scores of 0.85, 0.81, and 0.72 for the entire tumor, tumor core, and enhancing tumor core, respectively, based on the BRATS 2013 dataset.

In 2015, Ronneberger et al. [5] provides a homogeneous fully CNN named U-Net. U-Net greatly increased effectiveness in medical image segmentation tasks, leaving the downsampling process to link with the feature graph of the downsampling process to capture contextual information and the correct upsampling process to establish the correct location. U-Net is frequently utilized in the sector of medical image analysis as a result of it being exceptionally efficient in the final training of a limited number of images.

Several researchers applied U-Net architectures to segment brain tumors. Deep convolutional networks based on U-Nets, as proposed in 2020 by Ramy et al. [6], and in 2017 by Hao Dong et al. [7], are used to build a fully programmed brain tumor recognition and segmentation technique. In 2020, Xue Feng et al. [8] produced a 3D U-Net ensemble for brain tumor segmentation, whereas in 2018, Kamnitsas et al. [9] presented EMMA (Ensembles of Multiple Models and Architectures) for vigorous brain tumor segmentation. In 2021, Sadad et al. [10] developed U-Net with ResNet50 architecture for segmentation of tumors utilizing the Figshare dataset, getting an IoU score of 0.9504.

In 2020, Gadosey et al. [11] introduce the stripped-down U-Net (SD U-Net), a deep neural network that is highly fast, compact, and computationally effective for segmenting medical images on devices with low processing resources. Although these approaches offer certain advantages in some areas, the problem of unequal tumor and background voxel distribution in brain tumor segmentation must be addressed immediately. As a result, this study presents a learning mechanism for improving the segmentation approach.

In 2020, for the segmentation of MRI brain tumors, Pravitasari et al. [12] proposed UNet-VGG16. This model or architecture is a fusion of the U-Net and VGG16 architectures, with transfer learning used to simplify the U-Net architecture. The learning dataset shows that this approach has a high accuracy of 96.1%. Calculating the correct classification ratio (CCR) and comparing the segmentation result with the ground truth are used to validate the segmentation result. With a mean CCR score of 95.69 %, this UNet-VGG16 was able to distinguish the brain tumor area.

In 2020, for brain tumor subregional segmentation, Lyu et al. [13] presented a two-stage encoder-decoder-based methodology. To avoid the overfitting problem, both stages use variational autoencoder regularisation. The second-stage network employs attention gates and is trained with a larger dataset derived from the first-stage outputs. The suggested method yields a mean Dice score of 0.9041, 0.8350, and 0.7958, and a Hausdorff distance (95 percent) of 4.953, 6.299, and 23.608 for the total tumor, tumor core, and enhancing tumor, respectively, on the BraTS 2020 validation dataset. On the BraTS 2020 testing dataset, the corresponding Dice score values are 0.8729, 0.8357, and 0.8205, and the Hausdorff distance values are 11.4288, 19.9690, and 15.6711.

3. Proposed System

To ensure the effectiveness of MRI image segmentation based on the human brain, we addressed a U-Net with an EfficientNet-B7 encoder. The process shown in FIGURE 2 is that of the proposed approach for segmentation. In brief, the brain

MRI images were administered and then used for training the segmentation model. The segmented image can be predicted using the trained model.
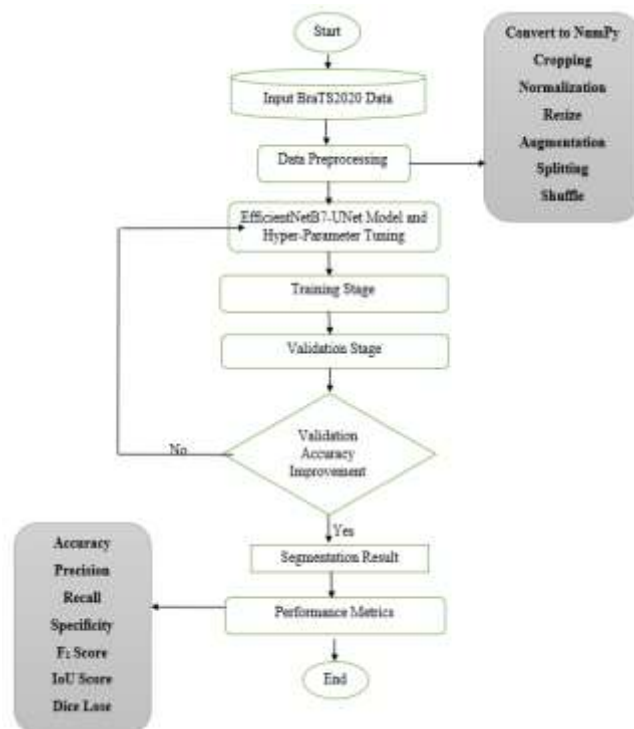

FIGURE. 2 FLOW CHART OF THR PROPOSED MODEL

3.1 U-Net Architecture

For the first time, Ronneberger et al. [5] (2015) suggested the U-Net architecture, a convoluted network designed only for biological-image analysis. The model is shaped like a "U". To put it simply, the encoder is a simple convolutional process, whereas its decoder is made up of 2D convolutional layers that have been transposed. The block diagram of U-Net architecture is depicted in FIGURE 3. The contraction route (also called the encoder component) is the primary route in the U-Net structure and is often used to capture the framework of the input image.

The encoder is made up of maximum pooling and convolution layers arranged one above the other. To permit precise localization using transposed convolutions, the second route is an uneven increasing route (additionally called the decoder component). Fully convolutional networks can only interpret images of a certain size since they have no thick layers, making them unable to handle larger images. Before max-pooling, two consecutive convolution layers are implemented in this type of architecture. The dimensions of the data are halved when pooling is applied, as a result, a lot of data will be lost. So, before each convolution, pooling layers are stacked to build up richer data representation without losing all spatial information quickly. To the U-Net structure, we added skip connections among the first convolution layer and the max-pooling layer on each level. The key purpose of this was to increase the accuracy and uniform distribution of parameters in the layers.
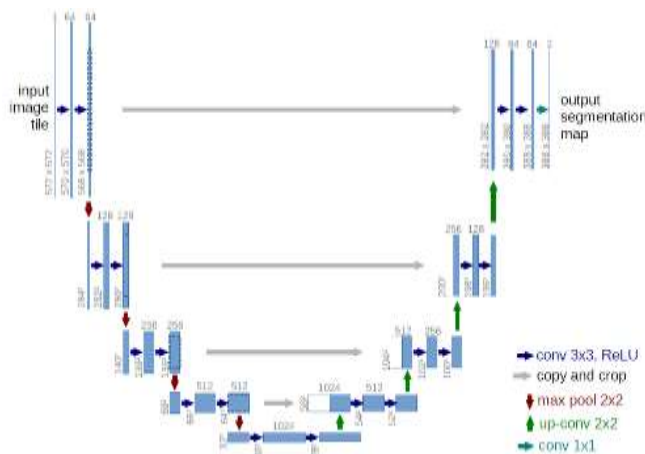

FIGURE. 3 U-NET ARCHITECTURE

3.2 Transfer Learning

Use a pre-trained ImageNet-trained CNN model for image classification instead of constructing a new CNN model for image classification in deep learning. Sinno Pan and Qiang Yang [14] developed a framework to help people grasp Transfer Learning. Rather than beginning the learning process from the start, transfer learning works with existing learning.

3.2.1. EfficientNet Architecture

The EfficientNet model was developed by Mingxing Tan and Quoc V. Le [15] of Google Research's Brain team, who published their findings in the journal "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks". To begin, a new baseline architecture named "EfficientNetB0" was constructed and then scaled up to build the EfficientNet family through a compound scaling method. Eight EfficientNet variations are powered by this method, ranging from 5.3 million parameters to 66 million.

The researchers began by automating the construction of neural networks by constructing a primitive network using a method known as the search for neural network architecture. It maximizes both accuracy and efficiency by calculating the number of FLOPS (floating-point operations per second). This design makes use of convolution with a moveable inverted bottleneck (MBConv). The amount of these MBConv segments varies according to the EfficientNet family. As we proceed through EfficientNet B0-B7, the depth, width, resolution, and model size all rise, while the accuracy also increases. The EfficientNetB7 model outperforms earlier state-of-the-art CNNs on ImageNet [15]. The complete architectural details of EfficientNet B7 are shown in FIGURE 4. The network begins with the stem layer and ends with the final layers. After the Stem layer, there are 7 blocks and each block contains different modules. The complete architecture diagram of U-Net with EfficientB7 is illustrated in FIGURE 5.
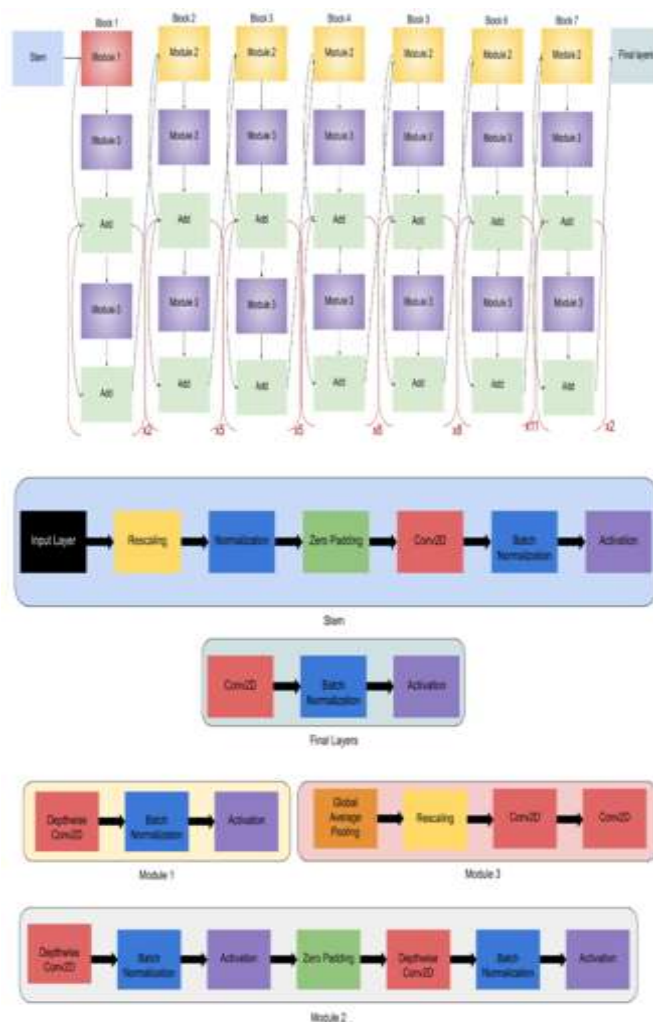

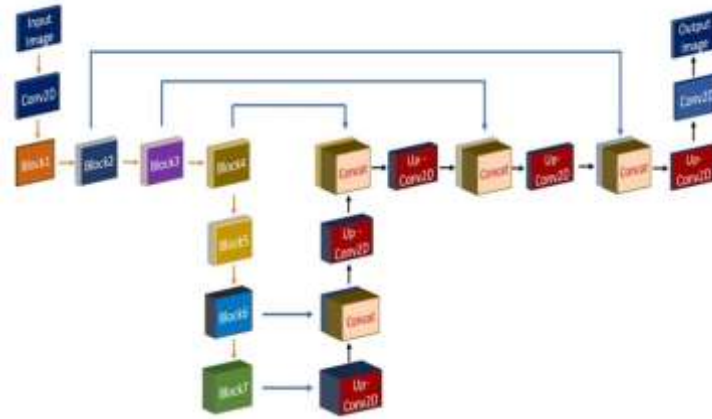FIGURE. 4 EFFICIENTNET-B7 COMPLETE ARCHITECTURE DETAILS

FIGURE. 5 U-NET WITH EFFICIENTNET-B7

3.3 Data Set

The publicly accessible benchmark dataset was utilized in this study. The BraTS 2020 dataset has been suggested for the identification and segmentation of brain lesions in an automated manner.

The Brain Tumor Segmentation Study (BraTS) has traditionally been designed to examine cutting-edge techniques for the segmentation of brain tumors in MRI. The BraTS2020 dataset was produced entirely from preoperative data collected from a variety of institutions, and it focuses on segmenting brain tumors that are fundamentally diverse (in terms of shape, histology, and appearance), including gliomas.

HGG (293) and LGG (76) instances are included in the BraTS 2020 training dataset. T1, T2, T1ce, and T2-FLAIR imaging modalities are co-registered with a voxel size of 248 × 248 and an isotropic resolution of one millimeter for all imaging modalities. The training data has annotations, but the validation and testing datasets (125 and 166 instances, respectively) do not. Estimated volumes of segmentation may be submitted to the organizer's website by participants to compare their methods. For the validation assessment, several contributions are permitted; however, for the final testing evaluation, only one submission is permitted per participant.

NifTI files with T1, T2, T1ce, and FLAIR descriptions are supplied for all multimodal BraTS scans. These scans came from diverse clinical procedures and scanners from various organizations. Images were manually segmented by one to four assessors for all of the datasets. There are many types of tumor cores, including necrotic and non-enhancing tumor cores, enhancing tumors, peritumoral edema, and others. The different modalities of images and masks are depicted in FIGURE 6.
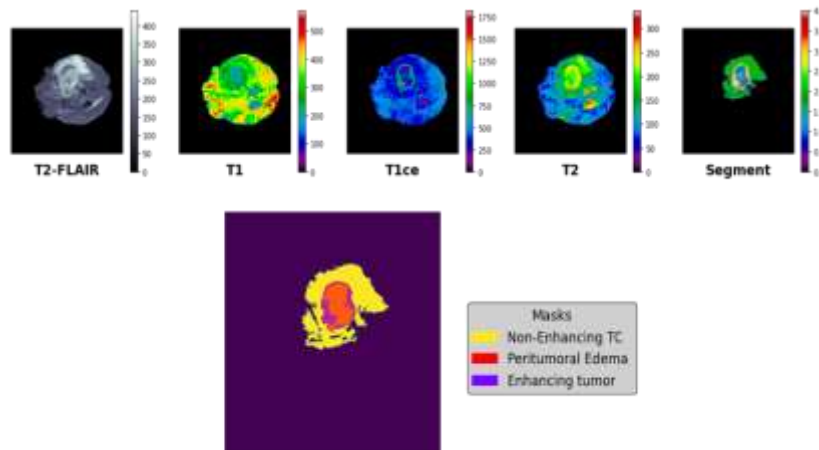

FIGURE 6. MULTI-MODAL IMAGES AND SEGMENTED MASKS

3.4 Data Preprocessing

Cropping images from the center is done by this method. Standardscaler normalization removes any distortion from the MRI intensity measurement, which is dependent on the imaging technology and scanner utilized.

The standard score for Sample x is computed in the following way:

$$Z = \frac{(x - u)}{s} \qquad (1)$$

"Where s is the standard deviation of the training samples, or 1 if it is False, and u denotes the mean of the training samples, or 0 if it is False. According to the standard scalers operating concept, the data will be transformed into distribution with a standard deviation of one and a mean of zero" [17]. This is done feature by feature in the case of multivariate data.

3.5   Training and Implementation details

The BraTS dataset contains MRI images and related segmentation output images for training and testing models. Clinically skilled neuroradiologists correct the output or ground truth labeled images. Folders for train and validation datasets are included in the BraTS dataset. All 369 subfolders in the train data folder are organized by modality, and each subfolder includes 5 nifty-format images of 5 distinct modalities. As a result, the train data folder has 1845 images. Each of the 125 subfolders in the validation data folder comprises four images of four distinct modalities, such as T1, T2-Flair, T2, and T1ce. As a result, there are a total of 500 images in the data folder for validation.

The dataset is split into HGG and LGG folders and each folder contains 5 modalities such as Flair, T1, T2, T1ce, Seg. The nifty images from each folder are loaded using the nib. load(). There are 155 slices of each nifty image and load each slice and apply some preprocessing such as image resizing, cropping, and standardization and save those images into Flair, T1, T2, T1ce, Seg folders as NumPy files using np. save(). The process is repeated until each folder contains 5000 NumPy files. After evaluating the data, the images are pre-processed by standardizing the intensity value and cropping them. MRI records are split into three categories: train set, test set, and validation set. The train set uses 70 % of the images, the test set uses 20 % and the validation set uses 10 %. The training images are data augmented, which aids generalization and enhances accuracy. Augmentation techniques are taken from the albumentation library such as grid distortion, random brightness contrast, elastic transform, optical distortion was employed to enhance the input image and offer more information for the model to learn. Before we can begin training our model, we need to build the learning process. This includes an optimizer, a loss function, and if desired, additional metrics such as F_1 score and IoU score. The Adam optimizer was used, with a learning rate of 0.0001. Each HGG and LGG image is trained separately.

U-Net with an EfficientNet-B7 encoder for both HGG and LGG images has also been employed in this study. In brain tumor representations, skip topologies provide perfect segmentation by using a high-level expression from dense sequencing layers. The three components of the design are bottleneck, contraction, and expansion. The expansion is made up of many contraction lengths. A convolution level addition input is followed by a maximum pooling limit for each block. In each block, each CNN layer has its sample level, with the lowest layer modulating between the expansion and contraction levels. After convolutions and activations, the ReLU activation function and batch normalization were utilized to prevent deep learning models from falling out of the experiment.

Features are extracted from the data loader to train the machine learning modes. The extracted features are then converted to a 2D array for better fitting the model.

3.6   Performance Metrics

3.6.1 Accuracy

Accuracy is a measure of the number of correct predictions out of all predictions [18].

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \qquad (2)$$

Here, we have true positives (TP) and negatives (TN), as well as false positives (FP) and negatives (FN).

3.6.2 Precision

Precision is a measure of the accuracy of a positive prediction. In other words, if an outcome is predicted to be positive, how certain that it can be actually positive [18].

$$Precision = \frac{TP}{(TP + FP)} \qquad (3)$$

3.6.3 Recall

The recall is the measure of how many true positives are predicted out of all actual positives in the dataset [19].

$$Recall\,(Sensitivity) = \frac{TP}{TP + FN} \qquad (4)$$

3.5.4 IoU Score

To compute the IoU score, divide the intersection point between the actual data (ground truth) and predicted segmentation by the point of union between the actual data (ground truth) mask and predicted segmentation mask." When assessing how much overlap there is between two masks or bounding boxes [20], it is a valuable statistic.

$$IoU = \frac{(ground\ truth \cap prediction)}{(ground\ truth \cup prediction)} \qquad (5)$$

3.6.5 F_1-Score

The F_1 score is determined by using "the harmonic mean of recall and precision" as the starting point [21]. The F_1 score is also the same as the dice score.

$$F - Score = 2 \times \frac{(precision \times recall)}{(precision + recall)} \qquad (6)$$

"An F1-score may have a highest possible value of 1.0, which shows perfect precision and recall, and a highest possible value of 0 if either precision or recall is zero".

3.6.6 Dice Loss

In this section, we'll talk about dice loss, which is computed as one minus the dice coefficient. The dice coefficient is a common metric for pixel segmentation that may also be used as a loss function if it is altered to work in this manner. " The dice coefficient is computed by multiplying the intersection area by the total number of pixels in both images" [22]. The dice coefficient is determined using the following formula:

$$Dice = 2 \times \frac{|A \cap B|}{|A| + |B|} \qquad (7)$$

$$Dice\ loss = 1 - Dice \qquad (8)$$

4. Results & Discussions

Using IoU, F_1 score, Dice Loss, Accuracy, Precision and Recall metrics, we evaluate the suggested architecture's segmentation capabilities. 200 epochs and 16 batch sizes are used to train the network. Four NVIDIA P40 GPUs are used to train our network in Pytorch.

The U-Net with the EfficientNet-B7 model is trained for both HGG and LGG images. The performance of the metrics of the model LGG is presented in Table I. The performance of the metrics of the model HGG is presented in Table II. When comparing the scores of U-Net with EfficientNet-B7 over HGG and LGG images, in the U-Net with the EfficientNet-B7 model for LGG images, the proportion of intersection between the targeted mask and our predicted output is higher when compared to HGG images.

The training parameters are the most important aspect to take into account when calculating the compute time of a CNN. Using the same dataset and setting up all training parameters, in the same manner, is thus critical. After the network has been trained, it may be used to image segmentation.

The total time taken for training the model is around 7 to 8 hours for both HGG and LGG images. Among the problems that we encountered during the design of the system is that we needed a GPU and a local machine with very high specifications for training, so we used to google colab and remembered the advantages of using it. In just a few seconds, the trained model can segment images and provide a final segmentation. Manual tumor segmentation by clinicians, on the other hand, might take many hours or even days. Accurate, rapid, and low-cost image segmentation approaches are recommended. There is a chance that this might save the lives of countless individuals by allowing physicians to quickly and precisely diagnose a brain tumor.

Figure 7 depicts the prediction of U-Net with EfficientNet-B7 for HGG images. Figure 8 depicts the prediction of U-Net with EfficientNet-B7 for LGG images.

TABLE I

PERFORMANCE METRICS FOR LGG DATA

Dice Loss	IoU	Precision Score	F₁ Score	Accuracy	Recall
			Training Results		
0.0331	0.8728	0.9280	0.9303	0.9892	0.9326
			Validation Results		
0.0367	0.8959	0.9438	0.9915	0.9505	0.9374

TABLE II

PERFORMANCE METRICS FOR HGG DATA

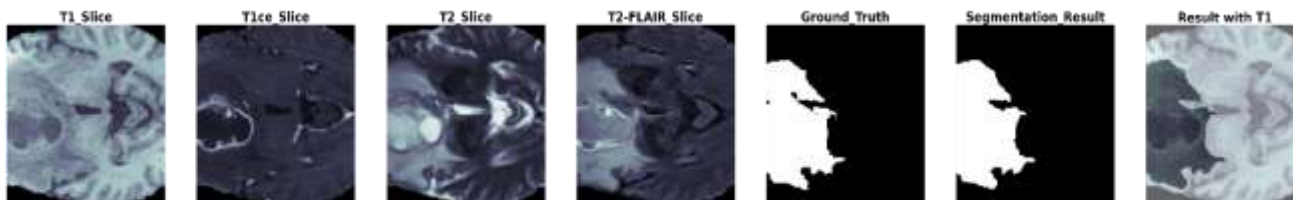| Dice Loss	IoU	Precision Score	F₁ Score	Accuracy	Recall
			Training Results		
0.0418	0.8531	0.9140	0.9182	0.9874	0.9225
			Validation Results		
0.0483	0.8346	0.9037	0.9938	0.9237	0.8899

FIGURE. 7 PRIDCTION BASED ON U-NET WITH EFFICIENTNET-B7 FOR HGG IMAGES
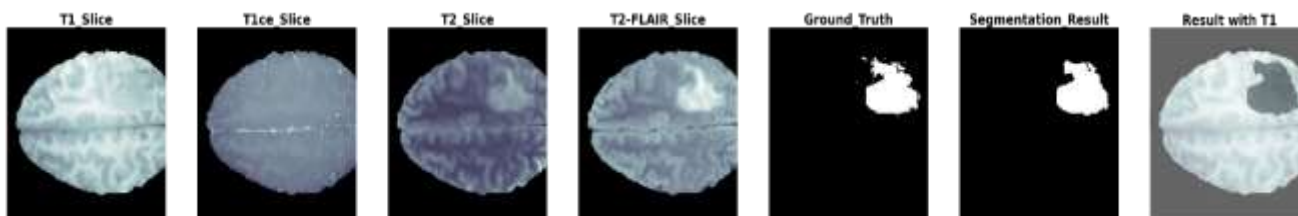

FIGURE. 8 PRIDCTION BASED ON U-NET WITH EFFICIENTNET-B7 FOR LGG IMAGES

5. Conclusion and Future Work

It may be challenging to segment brain tumors due to the intricacy of MRI brain imaging; nonetheless, its objectives to foresee malignancies via the use of artificial intelligence models make this effort worthwhile. For the automatic segmentation of brain tumors, the suggested system makes use of a U-Net with EfiicientNetB7 architecture. They simplify and expedite the imaging and segmentation of brain tumors. The U-Net with EfficientNetB7 architecture for LGG images IoU score was 0.8728 according to the results of the tests, the U-Net with EfficientNet-B7 design for LGG images plays a critical role in the treatment of brain tumors, particularly in the early stages. It offers a framework for predicting the segmentation of brain lesions and aids in the precise segmentation of the location of the lesions, both of which are important. Our study reveals that the suggested technique outperforms the current approaches for the segmentation of brain tumors by a significant margin.

MRI data cannot be completely exploited by the 2D U-Net model, hence the architecture lacks semantics and local characteristics between slices due to the model's restrictions. A 3D network model could be used in the future to increase performance and demonstrate the architecture's generalizability by applying it to different datasets. Because of this, it is still difficult to use deep neural networks for the segmentation of brain tumors because of the intricacy of brain MRI imaging and the scarcity of labeled data. As a result, we'll be working on a medical brain tumor segmentation model in the future.

The study could be expanded in the future by developing more powerful patch extraction techniques to help improve segmentation accuracy. So far, we have only introduced the use of 2D patches for extraction and training; however, this work can be extended to include the creation of models to be trained on 3D patches to find effective ways to extract 3D patches. In the future, we would want to work on different transfer learning models as an encoder architecture for comparing the performance of the segmentation model. We would also want to work on different pre-processing techniques for better segmentation. In addition to the prediction of tumors, in the future, we will also predict the overall survival of patients using radiomic features.

References

[1] "Brain Tumor: Types, Risk Factors, and Symptoms." https://www.healthline.com/health/brain-tumor (accessed May 25, 2021).

[2] Atiyah, A. Z., & Ali, K. H. (2021). Brain MRI Images Segmentation Based on U-Net Architecture.

[3] F. Isensee, P. F. Jäger, P. M. Full, P. Vollmuth, and K. H. Maier-Hein, "nnU-Net for Brain Tumor Segmentation," pp. 118–132, 2021.

[4] Le, Hai & Hien, Pham. (2018). Brain tumour segmentation using U-Net based fully convolutional networks and extremely randomized trees. Vietnam Journal of Science, Technology and Engineering. 60. 19-25. 10.31276/VJSTE.60(3).19.

[5] O. Ronneberger, Philipp Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," CoRR, vol. abs/1505.0, pp. 16591–16603, 2015.

[6] R. A. Zeineldin, M. E. Karar, J. Coburger, C. R. Wirtz, and O. Burgert, "DeepSeg: deep neural network framework for automatic brain tumor segmentation using magnetic resonance FLAIR images," Int. J. Comput. Assist. Radiol. Surg., vol. 15, no. 6, pp. 909–920, 2020.

[7] H. Dong, G. Yang, F. Liu, Y. Mo, and Y. Guo, "Automatic brain tumor detection and segmentation using U-net based fully convolutional networks," Commun. Comput. Inf. Sci., vol. 723, pp. 506–517, 2017.

[8] X. Feng, N. J. Tustison, S. H. Patel, and C. H. Meyer, "Brain Tumor Segmentation Using an Ensemble of 3D U-Nets and Overall Survival Prediction Using Radiomic Features," Front. Comput. Neurosci., vol. 14, no. April, pp. 1–12, 2020.

[9] K. Kamnitsas et al., "Ensembles of multiple models and architectures for robust brain tumour segmentation," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 10670 LNCS, pp. 450–462, 2018.

[10] T. Sadad et al., "Brain tumor detection and multi-classification using advanced deep learning techniques," Microsc. Res. Tech., no. October 2020, pp. 1296–1308, 2021.

[11] P. K. Gadosey et al., "SD-UNET: Stripping down U-net for segmentation of biomedical images on platforms with low computational budgets," Diagnostics, vol. 10, no. 2, pp. 1–18, 2020.

[12] Pravitasari, A. A., Iriawan, N., Almuhayar, M., Azmi, T., Fithriasari, K., Purnami, S. W., & Ferriastuti, W. (2020). UNet-VGG16 with transfer learning for MRI-based brain tumor segmentation. Telkomnika, 18(3), 1310-1318.

[13] Lyu, C., & Shu, H. (2020). A Two-Stage Cascade Model with Variational Autoencoders and Attention Gates for MRI Brain Tumor Segmentation. arXiv preprint arXiv:2011.02881.

[14] S. Panigrahi, A. Nanda, and T. Swarnkar, "A Survey on Transfer Learning," Smart Innov. Syst. Technol., vol. 194, pp. 781–789, 2021.

[15] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," 36th Int. Conf. Mach. Learn. ICML 2019, vol. 2019-June, pp. 10691–10700, 2019.

[16] B. Baheti, S. Innani, S. Gajre, and S. Talbar, "Eff-UNet: A novel architecture for semantic segmentation in unstructured environment," IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work., vol. 2020-June, pp. 1473–1481, 2020, doi:

10.1109/CVPRW50498.2020.00187

[17]   "sklearn.preprocessing.StandardScaler — scikit-learn 1.0.1 documentation." https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html (accessed Dec. 01, 2021).

[18]   "How to Calculate Precision, Recall, and F-Measure for Imbalanced Classification." https://machinelearningmastery.com/precision-recall-and-f-measure-for-imbalanced-classification/ (accessed Dec. 01, 2021).

[19] "Accuracy, Recall, Precision, F-Score & Specificity, which to optimize on? | by Salma Ghoneim | Towards Data Science." https://towardsdatascience.com/accuracy-recall-precision-f-score-specificity-which-to-optimize-on-867d3f11124 (accessed Dec. 01, 2021).

[20] "Intersection over Union (IoU) for object detection - PyImageSearch." https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/ (accessed Jul. 16, 2021).

[21] "F-Score Definition | DeepAI." https://deepai.org/machine-learning-glossary-and-terms/f-score (accessed Jul. 16, 2021).

[22] "An overview of semantic image segmentation." https://www.jeremyjordan.me/semantic-segmentation/ (accessed Jul. 16, 2021)