Multimedia Tools and Applications (2023) 82:10723–10743
https://doi.org/10.1007/s11042-022-13730-1

Analysis of depth variation of U-NET architecture for brain tumor segmentation

Biswajit Jena[1] · **Sarthak Jain**[1] · **Gopal Krishna Nayak**[1] · **Sanjay Saxena**[1]

Received: 30 December 2020 / Revised: 28 May 2021 / Accepted: 25 August 2022 /
Published online: 12 September 2022

Abstract
U-NET is a fully convolutional network (FCN) architecture designed to research the segmentation of biomedical images. The depth of the U-NET is one of the major constraints of this model while computing the performances. The larger depth of the U-NET means that its computational complexity is high as well. In certain cases, this large depth, as in the original model, is not justified for biomedical imaging modalities. In this paper, we have done an efficient analysis of U-NET architecture's depth variation, i.e., after removing different layers. For the analysis, the datasets BraTS-2017 and BraTS-2019, which consist of High-Grade Glioma (HGG) and Low-Grade Glioma (LGG) MR Scans, have been used for tumor segmentation. We have achieved a dice coefficient of at least 0.8866 and as high as 0.8887 on the discovery cohort, and at least 0.8895 and as high as 0.8911 cross-validation replication cohort. The results show that there are the least significant changes occurring in the performance parameters while moving from the higher to the lower depth of the model. Hence, in this paper, we presented that the large depth of U-NET, which costs more in terms of computational complexity, is not always required. Moreover, the U-NET models with depth reduction, which decreases the computational complexity, can achieve nearly the same results as in the case of the original U-NET.

Keywords U-NET architecture · Fully convolutional network (FCN) · Brain tumor segmentation · Biomedical image segmentation · Deep learning · Convolutional neural network

1 Introduction

Glioma is the most dreadful type of brain tumor. In common, it is in benign condition with low-grade glioma (LGG), and the life expectancy of a patient suffering from this may be several years. It becomes malignant in nature with high-grade glioma (HGG), and the duration of life reduces, at best, expected to live about two years. The usual method for treating this includes chemotherapy and radiation, along with surgery. But these methods

✉ Biswajit Jena
c118002@iiit-bh.ac.in

[1] International Institute of Information Technology, Bhubaneswar, India

are incomplete, i.e., even if it helps to stop the growth of the tumorous tissues, but still unable to completely remove it physically [7, 29, 38].

However, treating the tumor is possible only if one can detect it, and this lies as a challenge yet. Only trained experts, i.e., neuro-radiologists, can detect a tumor. But brain tumor has become a genuine global health issue. And not everywhere and always can these neuro-radiologists be available for the detection of it. Also, in the present day, the method used to detect brain tumors has many steps, and the most common steps involve biopsy, neurological exam, MRI scan, CT scan, angiogram, and spinal tap. Undergoing so many steps can make it a lengthy and time taking process. Hence, automation of this detection process is highly required because, by the right tools, this automation can be performed without the presence of an expert and also speed up the process of detecting the tumor.

The noninvasive magnetic resonance imaging (MRI) technology has emerged as a front-line diagnostic tool for brain tumors without radiation. It renders anatomical picturization of the brain. Even more, MRI images for brain tumor segmentation have eased and fastened the process of detecting and diagnosing brain tumors [12, 15, 33, 35].

Over the year, various segmentation techniques are proposed for image segmentation, which partitions the image and then analyses the portion of the image at a much more granular level. Essentially, tumor detection and segmentation belong to the task of semantic segmentation. The most traditional segmentation techniques of digital image processing include region-based segmentation [2], edge-based segmentation [13], and clustering-based segmentation [8], are used for various segmentation tasks. All these segmentation techniques have their own pros and cons. With the success of machine learning and deep learning, the image segmentation task became much easier, at the cost of heavy training time. The segmentation methodologies such as Mask R-CNN [24], U-Net [32], SegNet [3], and DeconvNet [18] are some good examples of it. When it comes to medical image segmentation and especially brain tumor segmentation from brain MRI scans, we should choose the best image segmentation technique.

U-NET is one of the most popular Fully Convolutional Networks (FCN) [25], a neural network model in which the input can be of any size, while the output is of the same size as the input when padding is used in the convolutional layers. An FCN is a neural network that consists of only convolutional layers, upsampling, and pooling layers without any fully connected or dense layers. Olaf Ronneberger et al. developed U-NET [32], which is a modification and an extension of J Long et al.'s [25] fully convolutional network. One of the applications where there is an extensive usage of U-NET is the segmentation of biomedical images. Olaf Ronneberger et al. used it for the same on biomedical images (cell tracking) for their research. It is considered to be achieving state-of-the-art segmentation results, while traditional ConvoNet models [20, 21] are good at classification and object detection tasks. U-NET is widely utilized for its applications in object-detection and image segmentation as it provides the following advantages [25, 32]:

1) It gives good computational efficiency. 2) It can be trained even with smaller datasets. 3) It can be trained from one end to the other. 4) It is widely utilized for biomedical image segmentation. The model derives its name from the fact that its architecture resembles the English alphabet 'U'. The framework of this model is based on the classic architecture of FCN. The original U-NET architecture uses 23 convolutional layers, as depicted in Fig. 1. U-NET advantages from its architecture of skip connections between various levels of the network. The main reason for using concatenation operation is to retain the characteristics of the original input data [11, 16, 32]. Many more variations of U-NET architecture have been developed for the various specific applications, some of them are C-UNET [17], SegNet [1, 3], DeepUNet [22], 3D-UNET [16], V-NET [16].

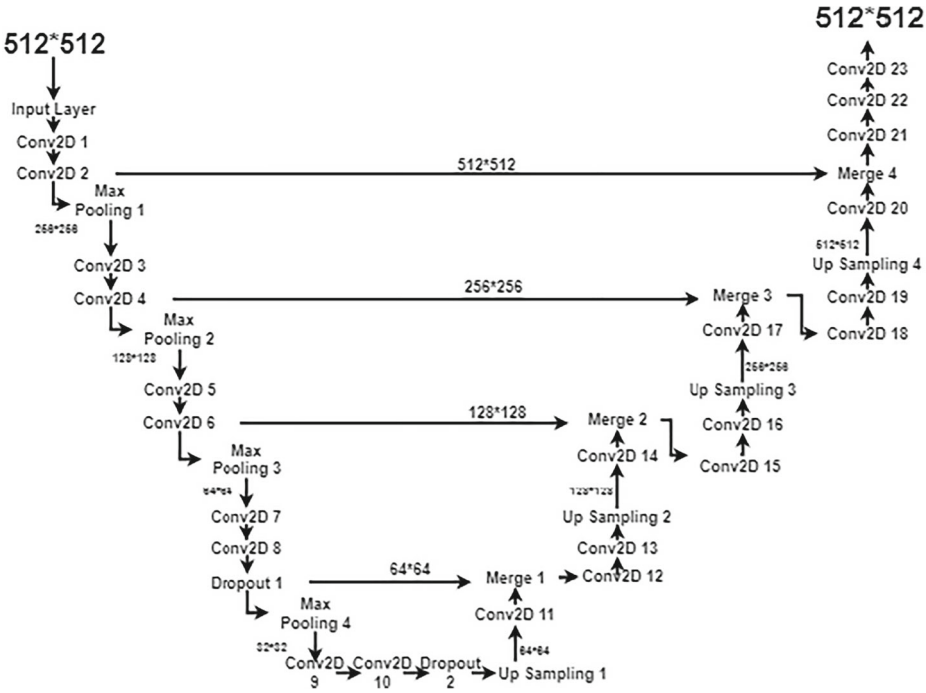

Fig. 1 The Original U-NET architecture

For the segmentation task, the U-NET classifies every pixel of the input image and gives the desired segmented result as output. In order to do so, the input image goes through two major sections of the U-NET, namely: 1) The contracting section and 2) The expanding section.

The contracting section consists of a sequence of convolutional and pooling layers. The convolutional layers help extract various patterns, which lay the groundwork for identifying the regions to be segmented from the original image. The pooling layers lower the size of the input images. Generally, max pooling is preferred for pooling operations. After every sequence of convolutional and pooling layers in the contracting section, the size of the output becomes half of the input size. A downward arrow, as shown in Fig. 1, illustrates this. By the end of the contracting path, the classification of all the pixels of the original input is completed [9, 14, 32, 38, 39].

In the convolutional neural network (CNN) case, the convolution is performed on the input image with the help of a kernel or filter to produce the feature map to make the output image size smaller. Therefore, deconvolution [6, 23, 28] came into the scenario to make the output size larger with upsampling. This is also referred to as up-convolution or transposed convolution or fractional stride convolution when the fractional stride is used.

In the expanding section, the resulted output from the contracting section goes for up-sampling to get a high-resolution segmented output. Each up-sampling (represented by an upward arrow in Fig. 1) increases the input size by a factor of two. Thus, at the end of the process, the resulting output is of equal size as the original input. The expanding section also consists of concatenations, which merges the outputs of the up-sampling layers to the

corresponding output of convolutional layers in the contracting section. This ensures a high-resolution segmented output at the end of the process. The reason for merging is that we may miss some features while increasing the size of the feature map by up-conv operation. To keep the feature intact, we copy and concat from the corresponding layer of the contraction path. So by this, we are getting the whole required feature map. Again after this, two convolution blocks consecutively operated to get a further feature map. So we are reducing these convo layers with a step of 2 in our proposed solution on each depth reduced model [9, 14, 31, 32, 39].

The summary and proposed solution of this article are as follow:

- The original U-NET model, as mentioned above, is expensive in terms of computation as compared to the depth-reduced models.
- In this paper, our proposed solution is basically for an extensive analysis of U-NET for depth reduction of the model.
- This allows the depth of reduced models to be trained quickly compared to the original model, which is required in the present day to start the treatment of the patient as quickly as possible.
- Hence in this article, we compare and analyze the depth-reduced models with the original U-NET model and see how they perform based on well know brain tumor segmentation datasets such as BraTS-2017 and BraTS-2019.
- Finally, this framework can be generalized to the applications of biomedical imaging diagnoses.

We organized the remaining section of this paper as follows. Related work is discussed in Section 3. Section 3 presents the proposed method, i.e., the architectural variation of the U-NET model. The datasets used for evaluation and the preprocessing tasks are explained in Section 4. Experimental setups with result analysis are put in Section 5. Finally, the discussion in Section 6, followed by conclusions in Section 7.

2 Related work

A variety of work has been done on brain tumor segmentation on multimodal images in the past few days. Most of the prominent work is based on magnetic resonance imaging (MRI) for automatic brain tumor segmentation, and some of the following shows the gist of the work.

The research paper by Kaus et al. [19], performed the automated brain tumor segmentation on 3D MRI images of 20 patients against manual segmentation and compared the time interval for both, in which the automated process takes only 5-10 minutes as compared to 3-5 hours in manual segmentation. From this, it is proved that automation for segmentation tasks on MRI scans is the ultimate choice.

Marcel et al.'s work in [30] is an automatic brain tumor segmentation framework using outlier detection, suggesting the presence of edema in MRI images as it is equally infectious as tumor core. They find it from the T2 MRI image intensities, where edema and core tumors arise in the abnormal regions.

A generative probabilistic model was introduced by Menze et al. [27], over traditional multivariate brain tumor segmentation. The estimation algorithm trains with the healthy tissue image and atlas of the lesion. The algorithm can differentiate the tumor border and the latent atlas from the image data.

The above literature is based on the digital image processing (DIP) and Machine Learning (ML) approach on brain MRI scans to segment brain lesions. Even if they are providing some good results for segmentation tasks, still their performance under the scanner. Next, with the success and trending of the convolutional neural network (CNN) on computer vision application, the researcher started the brain tumor segmentation task using CNN. The next kinds of literature are based on the CNN approach on brain MRI scans.

In [29], the use of a convolutional neural network (CNN) on BraTS 2013 and 2015 datasets for brain tumor detection was proposed by Pereira et al. In their proposed method, image pre-processing, patch extraction, and use of 3x3 kernel were emphasized. Apart from this, the Dice similarity coefficient for both BraTS 2013 and BraTs 2015 is computed. The literature in [15] by Havaei, Mohammad, et al. is a fully automatic brain tumor segmentation process using deep neural networks, in which four modalities of the BraTS 2013 dataset were used with two-phase training and validation. The CNN model used was able to capture both local features and global contextual features at the same time. The final layer i.e., a convolutional representation of a fully connected layer that gives a 40 fold speed up and a cascading architecture of CNN, was used in which the output of one basic model was input to the subsequent models.

Finally, we studied and analyzed some literature based on the U-NET architecture, especially for brain tumor segmentation, as this architecture is an FCN and benchmark for biomedical image segmentation.

Ronneberger et al. in [32] take the help of U-NET architecture to perform segmentation on light microscopic images. In their work, the network and training strategy uses data augmentation to maintain the invariance and robustness of the network, when only a few training samples are available. As a whole, the network has 23 convolutional layers and uses GPU, images of size 512x512. For their work, they won the ISBI cell tracking challenge 2015. Dong, Hao, et al. [9] developed a U-NET for automatic segmentation of brain tumors and evaluated their model with multimodal MRI brain images of the BraTS-2015 dataset that have 220 HGG and 54 LGG cases. Their work was basically on efficient determination and segmentation of tumorous cells and sub-tumor regions. The data augmentation technique was also applied for better network performance and segmentation accuracy.

Hasan et al. [14] upgrade the U-NET architecture by substituting the deconvolution parts accompanied by an up-sampling with the nearest neighbor algorithm and also deployed an elastic transformation to amplify the training database that provides the architecture with robustness, especially for the segmentation of LGG. The suggested nearest neighbor Resampling based Elastic-Transformed (NNRET) U-NET deep neural framework has been trained on 285 glioma patients of the BraTS 2017 dataset. The model was also tested on 146 patients using the Dice similarity coefficient (DSC) and Intersection over Union (IoU) assessment parameters and replacing the classic U-NET architecture with betterment.

Yang et al. [38] proposed a modified CNN architecture of U-NET. The original U-NET was modified by combining it with the ResNet model to enhance the brain tumor segmentation job performance. Then the BraTS 2017 dataset was used to train and test the newly proposed model. In the validation set, attain average dice scores of 0.883, 0.781, and 0.748 for the whole tumor, tumor core, and enhancing tumor, respectively, and in training set with values of 0.877, 0.774, and 0.757, respectively.

Beers et al. [7], proposed a sequential 3D U-NET for brain tumor segmentation in which the structural nature of glioma is taken into consideration. To detect and segment edema, non-enhancing tumors, and enhanced tumor regions separately, they trained several 3D U-NET architectures based on convolutional neural networks. Initially, to predict the whole tumor along with edema, a certain type of training was organized, and its result then became

the input to the different models to predict enhancing and non-enhancing tumors. For the whole tumor, enhancing tumor and the tumor core, they achieve Dice scores of 0.88, 0.73, and 0.73, respectively, on the BraTS dataset.

Colmeiro et al. [31] also proposed a model based on the 3D U-NET to segment the brain tumor. The benefit of the 3D U-NET is that it extracts 3d features from all adjoining voxels. For coarse and fine segmentation architecture, the training dice value obtained was 0.85 and 0.9, respectively. They resulted in validation and testing dice scores for the whole tumor class were 0.86 and 0.82, respectively.

From the studies of the literature survey, it has been observed that most of the literature focuses on the conventional deep learning models for biomedical image analysis and segmentation. Also, few researchers try to make some variation over conventional deep learning models to exploits better results in the cost of computational complexities. Even if U-NET was originally developed for biomedical image segmentation and gets significant results for cell and other biomedical image datasets, we want our proposed model, i.e., depth-reduced U-net, on the BraTS dataset. So, keeping all these in mind, we have studied and implemented the U-NET on the BraTS dataset, then depth variation of U-NET with different hyper-parameters settings.

3 Architectural variation of U-NET

The objective of our proposed solution is to reduce the depth of the U-NET model, as depth is the major constraint while computing the performances. Having several layers increases the depth of the U-NET model, which in turn increases the computational complexity.

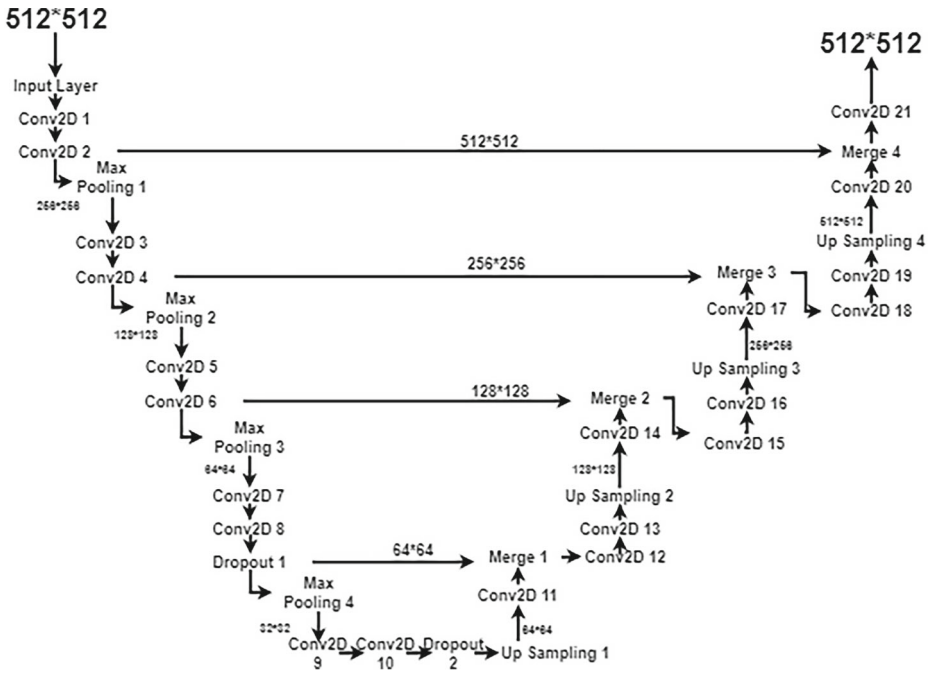

Fig. 2 Modified U-NET with 21 convolutional layers

Hence, more time is required to train the model. In this proposed solution, we analyze the U-NET's performance in which we have been reducing the depth of the model compared to the original one while keeping other optimized hyperparameters as of the original model. We analyze these depth-reduced U-NET models based on performance metrics like Dice Similarity Coefficient and Cross-Entropy Loss.

We have the original U-NET [32], which consists of 23 convolutional layers in total in the contracting path and the expanding path. The contracting path has 5 different convolutional blocks. Each block has 2 convolutional layers with a kernel size of 3×3, a stride of 1 in both directions, and a ReLU activation function. For the down-sampling, max pooling with kernel size 2×2 is used at the end of every convolutional block except the last block. While, in the up-sampling path, every block starts with a de-convolutional layer with a filter size of 2×2 and stride of 2, so that the size of feature maps increases twice.

Now, as we move into various architectures, we reduce the U-Net depth from the original architecture to the next architecture. As shown in Fig. 2, we reduce two convolution blocks from up-sampling paths; hence a total number of 21 convolutional layers are present in the second architecture. Similarly, we reduce the number of convolution layers in the subsequent models to make it 19, 17, 15, and 13 in corresponding models, as shown in Figs. 3, 4, 5 and 6. However, in the last architecture, apart from removing convolutional layers in the up-sampling path, we reduce it from the down-sampling path.

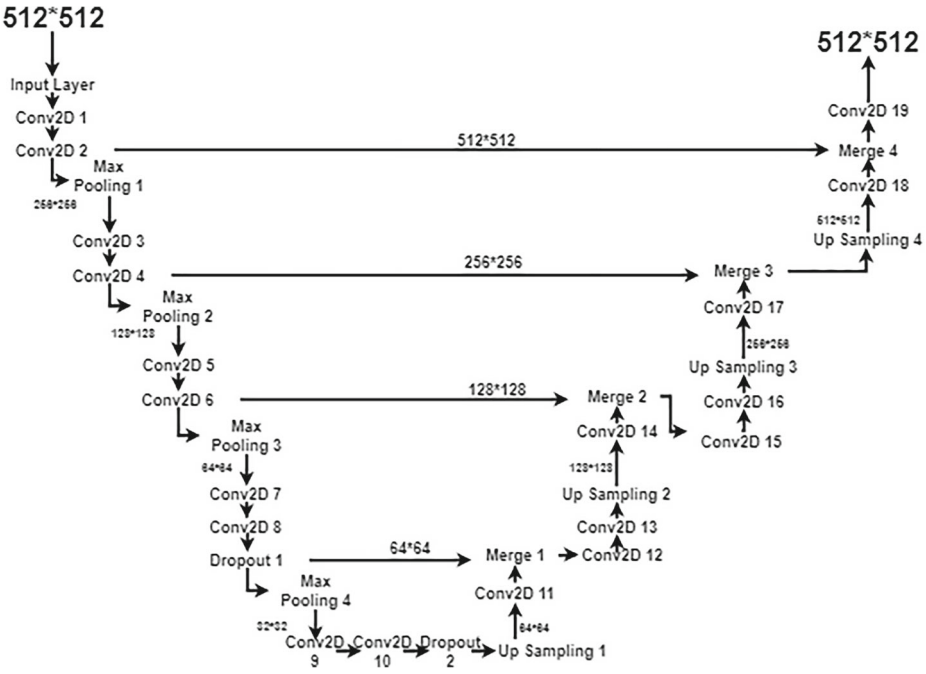

Fig. 3 Modified U-NET with 19 convolutional layers

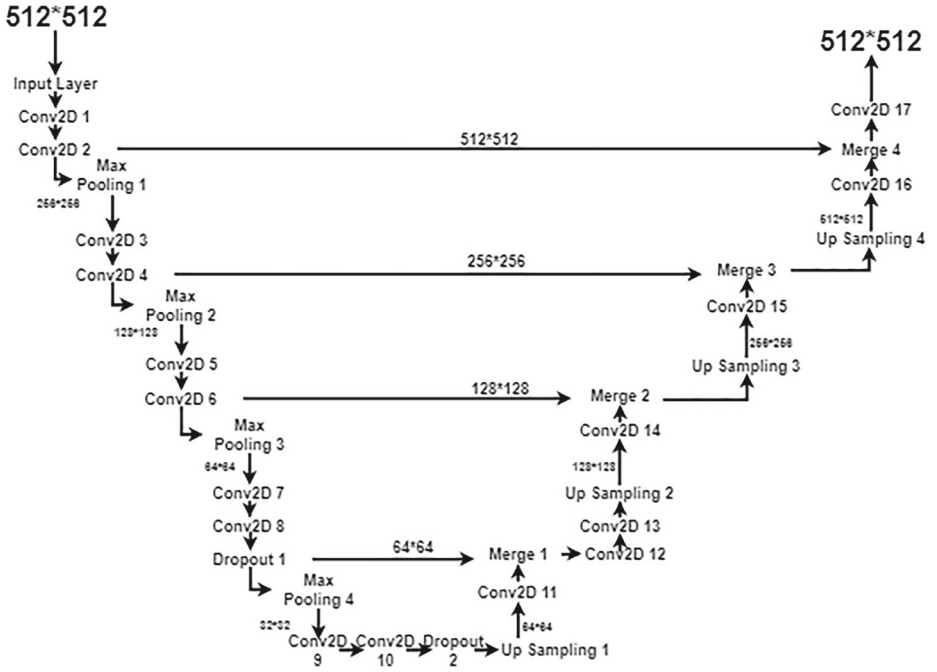

Fig. 4 Modified U-NET with 17 convolutional layers

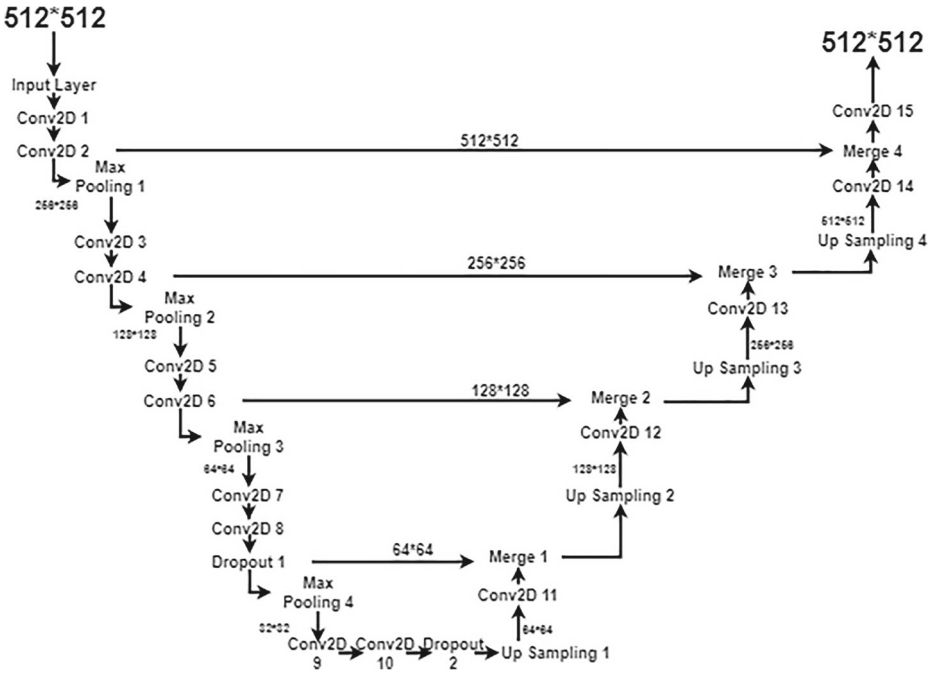

Fig. 5 Modified U-NET with 15 convolutional layers

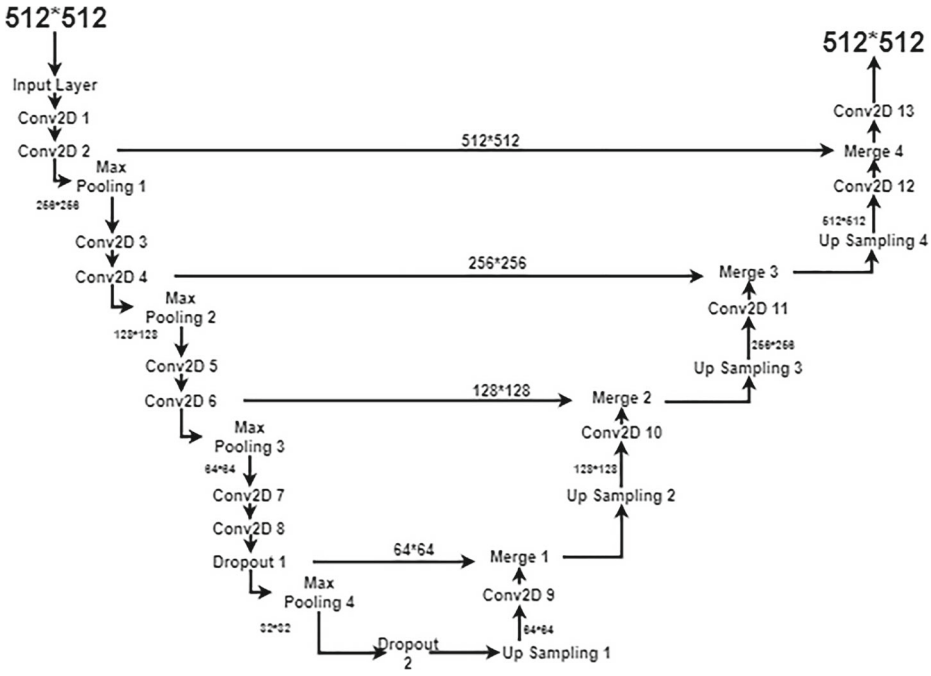

Fig. 6 Modified U-NET with 13 convolutional layers

4 Dataset and preprocessing

4.1 Dataset

The BraTS (Brain Tumor Segmentation) dataset has consistently been the center of attention on the assessment of state-of-the-art methodology for the segmentation of brain tumors in multimodal Magnetic Resonance Imaging (MRI). Our dataset for the segmentation of brain tumors was taken from the BraTS challenges [4, 5, 26, 27, 34]. BraTS conduct challenges every year, where the challengers are asked to develop deep learning-based models for automatic classification and segmentation of brain tumors from brain MRI scans. The training datasets after the completion of the challenge are freely available for research purposes.

The dataset consists of multimodal brain MRI scans along with manually annotated tumor regions corresponding to each scan in four different volumes, namely a) Native T1 (T1), b) Contrast-enhanced T1-weighted (T1ce), c) T2-weighted, d) Fluid Attenuated Inversion Recovery (Flair). Samples of all these modalities of the images are shown in Fig. 7. The scans were accumulated from actual cases of low-grade as well as high-grade glioma. Expert neuro-radiologists manually performed the segmentations. Corresponding to every patient, each of the volumes consists of 155 slices of 2D images, each having dimensions of 240 x 240, representing the 3D architecture of the brain. For our analysis, we considered

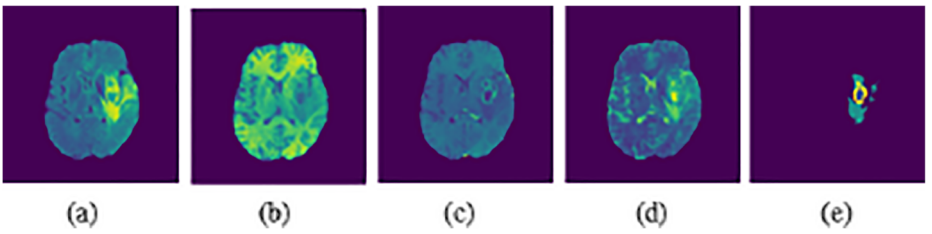

Fig. 7 Middle slices of an HGG patient in a) Flair b) T1 c) T1ce d) T2 e) segmented ground truth

both the HGG and LGG of flair volumes of BraTS-2017 and BraTS-2019 datasets [4, 5, 26, 27, 34].

4.2 Preprocessing of BraTS dataset

BraTS 2017 and 2019 contains 155 slices for every modality of each patient as it is a 3D dataset and is originally provided in .nii (NIfTI) format. We chose 5 slices out of 155 slices from each patient's data. These 5 slices were chosen to make sure that the brain's images present in them were not small. The 5 slices were chosen carefully, such as they were at least 3 or 4 slices away from each other. This takes care that different parts of the brain are covered, and the probability of finding a tumor becomes high. Each of the chosen slices was converted to .png (Portable Network Graphic) format as it eased the reading of images for the next preprocessing steps. The corresponding segmented ground truth images of each slice were also taken and saved in .png format. We applied data augmentation to these images.

The data augmentation steps included horizontal flips, vertical flips, width shift, height shift, rotation, shear, and zoom. The augmentation expanded the data about 5 times as we have to train the neural network and maintain the invariance and robustness of the network. Each image has dimensions of 240 x 240 was transformed into dimensions of 512 x 512 to match the U-NET model's input. The images were converted to grayscale, where the background was made white, and the brain tumor images were darker than the background, as shown in Fig. 8. The pixels that were very close to white were converted to complete white to increase the image's sharpness. Later, N4ITK bias field correction [1, 37] was applied to these images, which helped sharpen the tumor's borders. Now, since we are segmenting the GD-enhancing tumor, we have to alter the segmented ground truth images. The darkest color, i.e., black, represent the GD-enhancing tumor in the original grayscale images. All the other parts of the tumor, i.e., the peritumoral edema and the non-enhancing tumor core represented by gray and light gray, respectively, are whitened.

5 Experimental setup and result analysis

5.1 Experimental setup

The U-NET architecture has been implemented using python 3.6.7 with the Tensorflow library and Keras API. The various runtime information is provided as follows: The epoch size was considered as 40. The batch size was taken as 200. We have considered Adam optimizer with an initial learning rate of 0.0001. We have utilized the free resources of Google

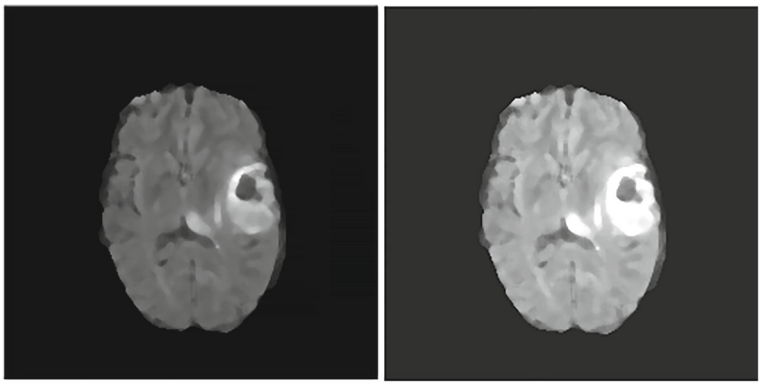

Fig. 8 A .png MRI flair scan (a) before and (b) after preprocessing jobs

Colab, which provides a free GPU to perform computations; however, the GPU memory is shared, as it is a cloud computing platform. The details of the GPU are: name: Tesla T4, major: 7, minor: 5, memoryClockRate(GHz): 1.59, pciBusID: 0000:00:04.0, totalMemory: 14.73GiB, freeMemory: 14.60GiB and compute capability: 7.5.

Most of the remote and unprivileged locations of the world do not have sufficient hardware resources. Here we exploit the free resources provided by Google Colab and work on a partial dataset, hence making a prototype and research initiative for full-fledged implementation on all volumes of the whole dataset and the latest dataset on the availability of the complete resource.

Finally, the pre-processed images were gone for 5-fold cross-validation steps in order to avoid overfitting of the model and fair training over all the data of the dataset.

Now, to measure the performance of the U-NET model for the image segmentation task against ground truth, the following metrics were chosen:

– **Dice similarity coefficient:** The Dice similarity coefficient (DSC) is a performance metric to calculate the accuracy of automatic image segmentation results from MRI scans by comparing it with the ground truth results. It is basically a statistical measurement metric, providing accuracy in a probabilistic manner. For two sets P and Q, it can be expressed as:

$$dice(P, Q) = 2 * |intersection(P, Q)|/(|P| + |Q|) \tag{1}$$

Where, $|P|$ and $|Q|$ represents the cardinal of sets P and Q, respectively.

– **Binary cross-entropy:** It is also known as log loss. It is one of the widely used activation functions. It measures the performance of a classification model. Its value varies between 0 and 1. Cross-entropy loss increases when the predicted probability of the dataset starts to vary from the actual label of the dataset. The cross-entropy can be measured as:

$$-\sum_{c=1}^{M} y_{o,c} log(p_{o,c}) \tag{2}$$

Where, M = number of classes, log = natural log y = binary indicator (values varies from 0 to1) if class label c is the correct classification for observation o p = predicted probability observation o is of class c.

We are using binary cross-entropy loss because we are doing binary segmentation to find the GD-enhancing tumor.

– **Accuracy:** It is the degree to which the result of a measurement or calculation conforms to the correct. In easier terms, it means how much the resulting data is closer to the original data.

$$Accuracy = (TP + NP)/(P + N) \tag{3}$$

Where, TP = True Positive, TN = True Negative P = condition Positive (the number of real positive cases in data) N = condition Negative (the number of real negative cases in data).

5.2 Result analysis

As shown in Fig. 1, the original U-NET has 23 convolutional layers. We reduced the convolutional layers step by step. To make an easy comparison, we show the results for 5 different stages of depth reduction. As the number of convolutional layers decreased, the variations in the performance of the models were recorded. Further, a comparison is drawn between the original U-NET's performance and the modified U-NETs with reduced depths at different levels, which are shown in Tables 1 and 2 for BraTS-2017 and BraTS-2019 datasets, respectively. The results obtained are further elaborately analyzed in the discussion section. From the result analysis, we can observe that the variation in the performance is nominal over the depth-reduced models and also, there is an increase in the training speed for the models on reducing the depth. The predictions generated are initially a bit unclear, but we get clear segmentation results on further reduction in the layers.

Till variation 5 in Table 1 shows a gradual improvement in the overall performance, as the convolutional layers are removed from the expanding section of the U-NET. However, in variation 6, where one convolutional layer is removed from the contracting section and the rest of the convolutional layers removed are from the expanding section of the U-NET, we observe a decline in the overall performance of the U-NET. Apparently, this shows that depth reduction in the expanding section of the U-NET proves fruitful for enhancing its performance, while depth reduction from the contracting section of the U-NET hinders its performance. The same scenario is also observed in Table 2 for the BraTS-2019 dataset.

Now, Table 3 summarized the assessment of our proposed segmentation technique with some typical recent publications based on U-Net architecture for brain tumor segmentation on brain MRI scans. We especially consider DSC as the quantified parameter for comparison, as it is considered the best parameter for evaluating segmentation tasks and also, most literature we studied follows it. For the comparison, the MICCAI Multimodal Brain Tumor Segmentation dataset is considered as the benchmark dataset, in which the results are for the combined tumor grade of HGG and LGG. However, as we are using the U-Net architecture, we also show the result of the original article of the U-Net model, which illustrates that U-Net providing the state-of-the-art result for electron microscopic image segmentation.

6 Discussion

As discussed earlier, one of the major drawbacks of U-NET is its depth. Although the U-NET performances for brain tumor applications are satisfactory, the time required to train the model over a dataset is more. Thus, to overcome this issue, we reduced the U-NET's depth by reducing the convolutional layers. By considering depth as the major parameter, we reduce 2 convolutional layers in each step as we moved to the subsequent models from

Table 1 Comparative analysis of the performance evaluation parameters corresponding to the depth-reduced models on the BraTS-2017 dataset

Model	Avg time per epoch(closet integer)	Loss	Accuracy	Dice Co-efficient	Val. Loss	Val. Accuracy	Val. Dice Co-efficient
Original U-NET(23 convolution layer)	437 sec	0.147	0.8864	0.888	0.1486	0.8789	0.8904
Original U-NET(21 convolution layer)	330 sec	0.158	0.8851	0.8866	0.1499	0.8897	0.8901
Original U-NET(19 convolution layer)	147 sec	0.156	0.8854	0.8866	0.14	0.8901	0.8895
Original U-NET(17 convolution layer)	143 sec	0.154	0.8864	0.8873	0.1388	0.8898	0.8911
Original U-NET(15 convolution layer)	138 sec	0.15	0.8872	0.8887	0.1345	0.8907	0.891
Original U-NET(13 convolution layer)	203 sec	0.153	0.8865	0.8875	0.1363	0.8901	0.8902

Table 2 Comparative analysis of the performance evaluation parameters corresponding to the depth-reduced models on the BraTS-2019 dataset

Model	Avg time per epoch(closet integer)	Loss	Accuracy	Dice Co-efficient	Val. Loss	Val. Accuracy	Val. Dice Co-efficient
Original U-NET(23 convolution layer)	501 sec	0.133	0.8912	0.8955	0.1319	0.8817	0.9191
Original U-NET(21 convolution layer)	444 sec	0.149	0.8943	0.8951	0.1439	0.8997	0.9099
Original U-NET(19 convolution layer)	356 sec	0.144	0.8922	0.8944	0.1201	0.9001	0.9011
Original U-NET(17 convolution layer)	300 sec	0.146	0.8974	0.8978	0.1084	0.9198	0.9100
Original U-NET(15 convolution layer)	287 sec	0.143	0.8739	0.8612	0.1495	0.8707	0.8709
Original U-NET(13 convolution layer)	306 sec	0.148	0.8779	0.8577	0.1355	0.8961	0.8891

Table 3 Assessment of proposed segmentation technique with other recently published deep learning-based methods

Name	Dataset	Tumor Grade	Model	Results(DSC or IOU)
Ronneberger et al. [32](original U-net model)	electron microscopic		U-Net	IOU 0.92
Dong et al. [9]	BraTS-2015	Combined (HGG+LGG)	U-Net	DSC 0.86
Hasan et al. [14]	BraTS-2017	Combined (HGG+LGG)	U-Net(NNRET)	DSC 0.87 & IOU 0.85
Yang et al. [38]	BraTS-2017	Combined (HGG+LGG)	U-Net(ResNet)	DSC 0.88
Colmeiro et al. [31]	BraTS-2017	As per BraTS-2017 challenge	3D U-Net	DSC 0.86
Beers et al. [7]	BraTS-2017	Combined (HGG+LGG)	3D U-Net(Sequential)	DSC 0.88
Proposed	BraTS-2017	Combined (HGG+LGG)	Depth reduced U-Net	DSC 0.89

Table 4 Parameters setting for the depth reduced U-NET models

Parameter	Value
Pooling	Max-Pooling
Activation	ReLU
Kernel size	3x3, 2x2, 1x1
Regularisation	L1,L2,Dropout
Number of Convolution Block	5
Number of De-convolution Block	5

the original model. The other parameters like activation functions, kernel size, pooling operation, etc., in the original model, are the optimal parameters that have been used. We have also used all the optimized hyperparameters in our proposed solution. If we use any other hyperparameter values, we are not getting the required result as the original model. For e.g., if we use average-pooling instead of max-pooling, we are not getting the required result. Our goal is to reduce the computational complexity while preserving nearly the same result as the original model, which can be feasible by reducing the depth as the parameter. Table 4 also listed the parameters used in the network.

This paper presented our depth variation of the U-NET architecture trained with flair volumes of the BraTS 2017 dataset for automatic brain tumor segmentation. So, in this work, we used flair images to segment the tumor region that has been proved to be effective in these studies [9, 10, 36]. Depth of the U-NET being an essential part, but its presence sometimes gives more computational overhead with respect to performance issues. Hence

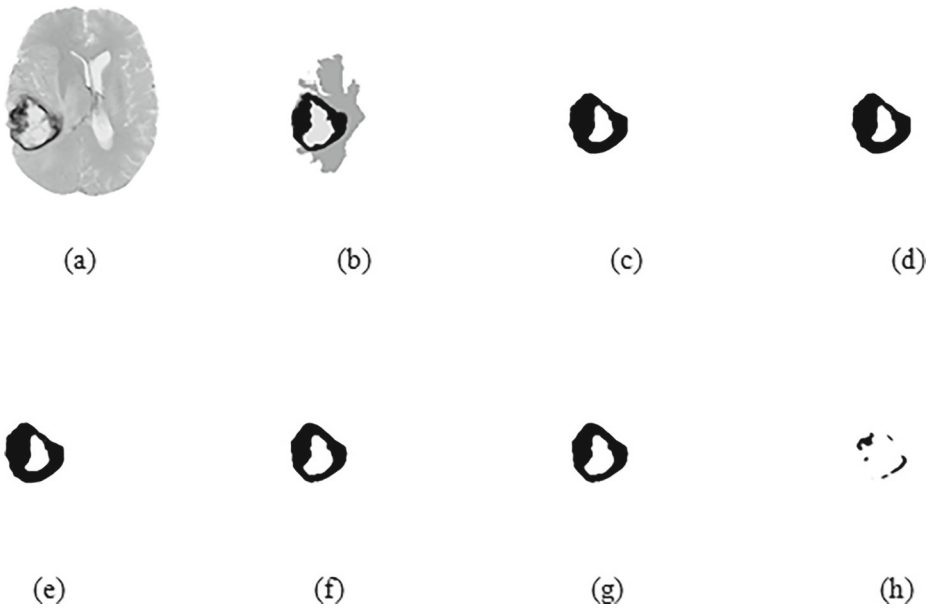

Fig. 9 (a) Example test image of BraTS-2017 dataset (b) Ground truth (c) Segmentation result of original U-NET (d) segmentation result of variation1 (e) segmentation result of variation 2 (f) segmentation result of variation 3 (g) segmentation result of variation 4 (h) segmentation result of variation 5

Fig. 10 (a) Same example test image corresponding of BraTS-2019 dataset (b) Ground truth (c) Segmentation result of original U-NET (d) segmentation result of variation1 (e) segmentation result of variation 2 (f) segmentation result of variation 3 (g) segmentation result of variation 4 (h) segmentation result of variation 5

reducing the depth of the architecture and, at the same time, maintaining the performance of the model is quite important. This procedure is followed by our proposed method, in which we reduce the depth in five consecutive steps. Each time, we train and validate the updated models with the same dataset and measure the performance parameters.

While we move from 1st depth reduction step to 5th reduction step, we are getting a better average execution time in each computation. The computational average epoch time difference between the 1st and 5th depth reduction step is an average of 300 seconds, which may be considered as a big difference concerning the GPU execution time. But as we make the depth reduction in the 6th step in the down-sampling path as shown in Fig. 6 (in all other cases, it's in the upsampling path), hence, only the time of computation bit higher in the 6th step as shown in Tables 1, 2 and Fig. 11, even with the degradation of the segmentation image, which is depicted in the resulted experimental diagram in Figs. 9(h), 10(h). This may be considered as the only obstruction of this proposed solution. The reason for reducing the layer from the up-sampling path is: In U-NET, we are doing a copy and concatenate operation (merging operation). The reason for merging is that we may miss some features while increasing the size of the feature map by up-conv operation. To keep the feature intact, we copy and concatenate them from the corresponding layer of the contraction path. So by this, we are getting the whole required feature map. Again after this, two convolution blocks consecutively operated to get further feature maps. So we are reducing these two convo layers with a step of 2 (Fig. 11).

For other performance metrics, we consider Accuracy and Dice Similarity Coefficient. We calculate the training and validation accuracy, loss, and Dice coefficient. The training accuracy is almost the same in all depth-reduced models, with a value of 0.88 with minor

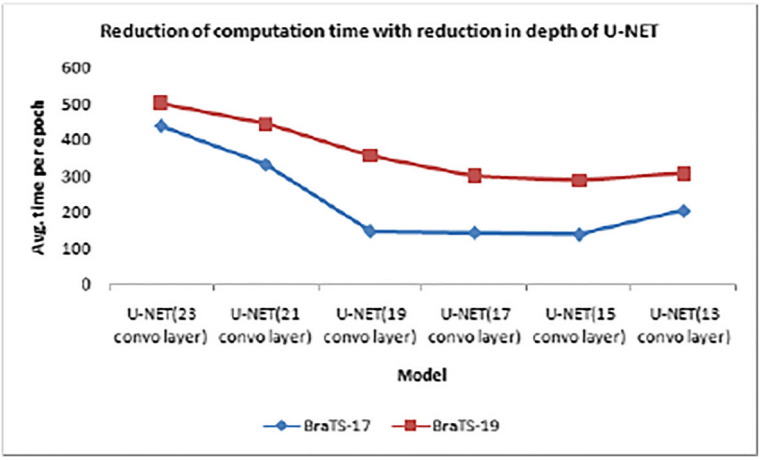

Fig. 11 Computational time variance graph as the depth of U-NET is reduced from 23 to 13 convolution layers

fractional changes. Rather, there is an improvement in validation accuracy from 0.87 to 0.89 from the original model to the last depth reduced model. Similarly, for Dice Similarity Coefficient, the training and validating values remain the same from the original model to the last model, as 0.88 and 0.89, respectively, shown on the graph Fig. 12. All these said results are from BraTS-2017. The same pattern of results also can be observed from BraTS-2019 as shown in the graph Fig. 13

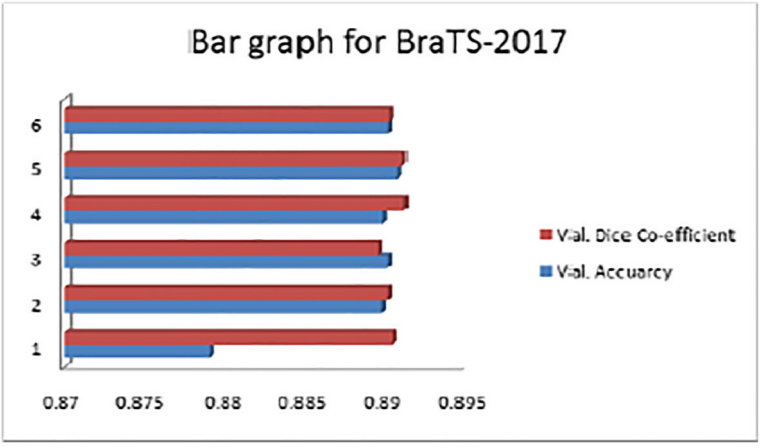

Fig. 12 Bar graph showing the validation accuracy and dice co-efficient values with depth-reduced U-NET models on the BraTS-2017 dataset

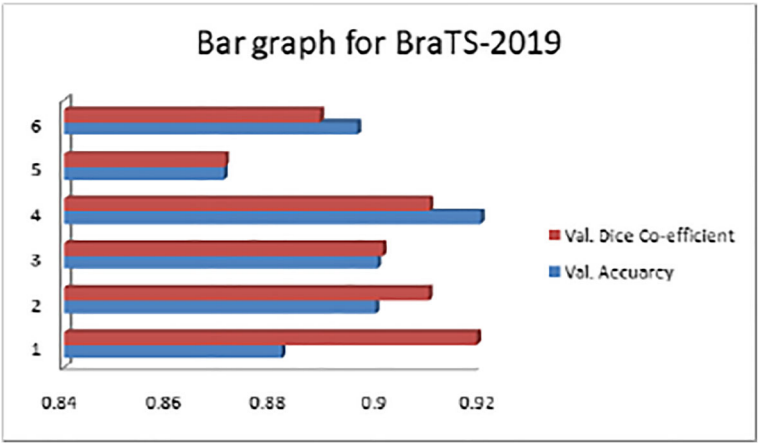

Fig. 13 Bar graph showing the validation accuracy and dice co-efficient values with depth-reduced U-NET models on the BraTS-2019 dataset

7 Conclusion

This paper presented a fully connected network (FCN), U-NET, with depth variation to segment the brain tumor on the BraTS 2017 and BraTS 2019 datasets, which gives remarkable accuracy and dice similarity co-efficient as performance measurement parameters over the classical U-NET model and with less computational complexity. Here, we conclude that depth reduction of the U-NET architecture has a crucial impact on enhancing the models' performance and hence simultaneously reducing the computational complexity overheads. However, the limitation came into the scenario when we are reducing the convolutional layers from the down-sampling path. By keeping all the pros and cons in mind, we can exploit the depth variation of the U-NET model to the recent BraTS datasets and other biomedical image datasets.

Funding The authors did not receive support from any organization for the submitted work.

Data Availability Yes

Code Availability Yes

Declarations

Conflict of Interests Authors declare that they have no conflict of interest.

References

1. Alqazzaz S, Sun X, Yang X, Nokes L (2019) Automated brain tumor segmentation on multi-modal mr image using segnet. Comput Visual Media 5(2):209–219
2. Astaraki M, Severgnini M, Milan V, Schiattarella A, Ciriello F, de Denaro M, Beorchia A, Aslian H (2018) Evaluation of localized region-based segmentation algorithms for ct-based delineation of organs at risk in radiotherapy. Phys Imaging Radiat Oncol 5:52–57

3. Badrinarayanan V, Kendall A, Cipolla R (2017) Segnet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans Pattern Anal Mach Intell 2481–2495(12):39
4. Bakas S, Akbari H, Sotiras A, Bilello M, Rozycki M, Kirby JS, Freymann JB, Farahani K, Davatzikos C (2017) Advancing the cancer genome atlas glioma mri collections with expert segmentation labels and radiomic features. Scientific Data 4:170117
5. Bakas S, Reyes M, Jakab A, Bauer S, Rempfler M, Crimi A, Shinohara RT, Berger C, Ha SM, Rozycki M et al (2018) Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge. arXiv:1811.02629
6. Baris K, Jensen G, van der Smagt P (2017) Cnn-based segmentation of medical imaging data. arXiv:1701.03056
7. Beers A, Chang K, Brown J, Sartor E, CP Mammen, Gerstner E, Rosen B, Kalpathy-Cramer J (2017) Sequential 3d u-nets for biologically-informed brain tumor segmentation, arXiv:1709.02967
8. Chowdhary CL, Mittal M, Pattanaik PA, Marszalek Z et al (2020) An efficient segmentation and classification system in medical images using intuitionist possibilistic fuzzy c-mean clustering and fuzzy svm algorithm. Sensors 20(14):3903
9. Dong H, Yang G, Liu F, Mo Y, Guo Yike (2017) Automatic brain tumor detection and segmentation using u-net based fully convolutional networks. In: Annual conference on medical image understanding and analysis, pages 506–517. Springer
10. Dvorak P, Bartusek K, Kropatsch WG (2013) Automated segmentation of brain tumour edema in air mri using symmetry and thresholding. PIERS Proc, Stockholm, Sweden
11. Ferdinand CP, Elshaer MEA, Ettlinger F, Tatavarty S, Bickel M Bilic P, Rempfler M, Armbruster M, Hofmann F, D'Anastasi M et al (2016) Automatic liver and lesion segmentation in ct using cascaded fully convolutional neural networks and 3d conditional random fields. In: International conference on medical image computing and computer-assisted intervention. Springer, pp 415–423
12. Gordillo N, Montseny E, Sobrevilla P (2013) State of the art survey on mri brain tumor segmentation. Magn Reson Imaging 31(8):1426–1438
13. Gupta D, Anand RS (2017) A hybrid edge-based segmentation approach for ultrasound medical images. Biomed Signal Process Control 31:116–126
14. Hasan SMK, Linte CA (2018) A modified u-net convolutional network featuring a nearest-neighbor resampling-based elastic-transformation for brain tissue characterization and segmentation. In: 2018 IEEE Western New York Image and Signal Processing Workshop (WNYISPW). IEEE, pp 1–5
15. Havaei M, Davy A, Warde-Farley D, Biard A, Courville A, Bengio Y, Pal C, Jodoin Pierre-Marc, Larochelle H (2017) Brain tumor segmentation with deep neural networks. Med Image Anal 35:18–31
16. Hesamian MH, Jia W, He X, Kennedy P (2019) Deep learning techniques for medical image segmentation: achievements and challenges. J Digital Imaging 32(4):582–596
17. Jiang H, Rong R, Junyan W, Li X, Dong X, Chen EZ (2018) Skin lesion segmentation with improved c-unet networks. BioRxiv:382549
18. Jun Fu, Liu J, Li Y, Bao Y, Yan W, Fang Z, Hanqing L (2020) Contextual deconvolution network for semantic segmentation. Pattern Recogn 107152:101
19. Kaus MR, Warfield SK, Nabavi A, Black PM, Jolesz FA, Kikinis R (2001) Automated segmentation of mr images of brain tumors. Radiology 218(2):586–591
20. Krizhevsky A, Sutskever I, Hinton GE (2017) Imagenet classification with deep convolutional neural networks. Commun ACM 60(6):84–90
21. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324
22. Li R, Liu W, Yang L, Sun S, Wei Hu, Zhang F, Wei Li (2018) Deepunet: a deep fully convolutional network for pixel-level sea-land segmentation. IEEE J Select Topics Appl Earth Observ Remote Sensing 11(11):3954–3962
23. Li X, Ren JS, Ce L, Jia J (2014) Deep convolutional neural network for image deconvolution. Adv Neural Inf Process Syst 27:1790–1798
24. Liu M, Dong J, Dong X, Yu H, Qi L (2018) Segmentation of lung nodule in ct images based on mask r-cnn. In: 2018 9th international conference on awareness science and technology (iCAST). IEEE, pp 1–6
25. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. Proc IEEE Conf Comput Vision Pattern Recognit:3431–3440
26. Menze BH, Jakab A, Bauer S, Kalpathy-Cramer J, Farahani K, Kirby J, Burren Y, Porz N, Slotboom J, Wiest R et al (2014) The multimodal brain tumor image segmentation benchmark (brats). IEEE Trans Med Imaging 34(10):1993–2024
27. Menze BH, Leemput KV, Lashkari D, Weber M-A, Ayache N, Golland P (2010) A generative model for brain tumor segmentation in multi-modal images. In: International conference on medical image computing and computer-assisted intervention. Springer, pp 151–159

28. Noh H, Hong S, Han B (2015) Learning deconvolution network for semantic segmentation. In: Proceedings of the IEEE international conference on computer vision, pp 1520–1528
29. Pereira S, Pinto A, Alves V, Silva CA (2016) Brain tumor segmentation using convolutional neural networks in mri images. IEEE Trans Med Imaging 35(5):1240–1251
30. Prastawa M, Bullitt E, Ho S, Gerig G (2004) A brain tumor segmentation framework based on outlier detection. Med Image Anal 8(3):275–283
31. Rodríguez CRG, Verrastro CA, Grosges T (2017) Multimodal brain tumor segmentation using 3d convolutional networks. In: International MICCAI brainlesion workshop. Springer, pp 226–240
32. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer, pp 234–241
33. Saxena S, Garg A, Mohapatra P (2019) Advanced approaches for medical image segmentation. In: Application of biomedical engineering in neuroscience. Springer, pp 153–172
34. Saxena S, Mohapatra P, Pattnaik Swati (2019) Brain tumor and its segmentation from brain mri sequences. In: Early detection of neurological disorders using machine learning systems, pp 39–60
35. Saxena S, Paul S, Garg A, Saikia A, Datta A (2020) Deep learning in computational neuroscience. In: Challenges and applications for implementing machine learning in computer vision. IGI Global, pp 43–63
36. Soltaninejad M, Yang G, Lambrou T, Allinson N, Jones TL, Barrick TR, Howe FA, Ye X (2017) Automated brain tumour detection and segmentation using superpixel-based extremely randomized trees in flair mri. Int J Comput Assist Radiol Surg 12(2):183–203
37. Tustison NJ, Avants BB, Cook PA, Zheng Y, Egan A, Yushkevich PA, Gee JC (2010) N4itk: improved n3 bias correction. IEEE Trans Med Imaging 29(6):1310–1320
38. Yang C, Guo X, Wang T, Yang Y, Ji N, Li D, Lv H, Ma T (2019) Automatic brain tumor segmentation method based on modified convolutional neural network. In: 2019 41st annual international conference of the IEEE engineering in medicine and biology society (EMBC). IEEE, pp 998–1001
39. Zhou X-Y, Yang G-Z (2019) Normalization in training u-net for 2-d biomedical semantic segmentation. IEEE Robot Auto Lett 4(2):1792–1799