2023 IEEE 7th Conference on Information and Communication Technology (CICT-2023)

Brain Tumor Segmentation using the Modified UNET Architecture

Harsh Garg, Ramavath Vasanth Naik, Tarun Harishchandra Pal, Mohit Kumar Mina, Anjali Gautam
Department of Information Technology, Indian Institute of Information Technology Allahabad(U.P.), India
harshgargckt@gmail.com, 12rvnaik@gmail.com, tarunpal13982@gmail.com, mohitkmina65@gmail.com, anjaligautam@iiita.ac.in

***Abstract*—Brain tumor is a life-threatening diseases and glioma is the most prevalent and dangerous type which can cause death if the tumor grade is high. Early detection of these tumors can improve and save the patient's life. Automatic segmentation of brain tumors from MRI scans plays a vital role in treatment planning and timely diagnosis. In simple UNET local features are lost during encoding or downsampling, resulting in constant learning as the model goes deeper. In this paper, we have used the spatial attention-based mechanism to help preserve local features. The modified architecture involves additional modules in the encoding path which are concatenated during upsampling or decoding. Our architecture achieved results with the average dice score for the tumor core (TC), whole tumor (WT), and enhancing tumor (ET) on the BraTS 2020 dataset of 0.9027, 0.8868, and 0.9067, respectively. To demonstrate the robustness of the proposed model in real-world clinical settings, validation of the trained model on an external cohort is performed on 50 percent of the BraTS 2021 benchmark dataset. The achieved dice scores on the external cohort are 0.8039, 0.7640, and 0.6689 for TC, WT, and ET, respectively.**

***Index Terms*—Brain tumor, Image segmentation, BraTS 2020 and 2021, U-Net, Attention Mechanism**

I. Introduction

Brain tumors include the most threatening types of tumors around the world [1]. A brain tumor is an abnormal growth of cells within the brain or the surrounding tissues that can cause a range of neurological symptoms and potentially life-threatening complications. Glioma, the most common primary brain tumor, occurs due to the carcinogenesis of glial cells in the spinal cord and brain. The brain tumors can be benign or malignant, and they can originate from different types of brain cells or from other organs that have spread to the brain. Therefore, they need special care during their treatment procedure. Machine learning algorithms have played an important role in the analysis of these tumors from imaging modalities. However, there are some limitations associated with getting the perfect boundary of the tumor region. The manual segmentation and analysis of structural MRI images of brain tumors is an arduous and time-consuming task and can only be accomplished by professional neuroradiologists [2]. Therefore, an automatic and robust brain tumor segmentation methodology needs to be present that can accurately detect the boundary of the tumor region. The accurate segmentation methodology will have a significant impact on brain tumor diagnosis and treatment. There are many different deep learning methods to segment MRI images but all methods didn't result in the most accurate segmentation.

In this paper, we have implemented 3D-UNET with Spatial Attention which gives better results than the normal 3D-UNET [4] and dRes-Unet 3D [5]. The problem with simple UNET is that during encoding or downsampling as we perform different operations due to which local features get lost and as we go deep in its layer architecture the learning becomes constant. So in order to avoid this loss of local features we use spatial attention [3], here, we have used the Res-UNET 3D architecture with some modifications. The encoding path involves some extra modules (min and max pooling) at each level, which helps in saving local features of which these modules are concatenated to the corresponding level during upsampling or decoding.

II. Related Work and LR

In this section, the work related to brain tumor segmentation is mentioned given. In previous research works, SAResU-Net [6], TransBTS [7], the hypercolumn technique-based model, the Spatial Attention-based Efficiently Features Fusion Network [8], and the Deep Neural Network-Based Novel Mathematical Model [9] have been developed for segmentation and many other as well. The findings suggest that a combination of different techniques, such as deep learning models, attention mechanisms, and multi-modal imaging, have shown promising results in improving brain tumor segmentation performance. The SAResU-Net combined shuffle attention and residual modules with a basic 3D U-Net and leveraged a self-ensemble module for improved performance. TransBTS used a transformer-based architecture to process multimodal medical images and generate tumor segmentation masks. The hypercolumn technique [10] based model employed attention modules and residual blocks for improved feature extraction and classification. The Spatial Attention-based Efficiently Features Fusion Network used a series of dilated multi-fiber units and spatial attention mechanisms for feature refinement and fusion. The Deep Neural Network-Based Novel Mathematical Model utilized a deep neural network for 3D brain tumor segmentation. The findings of the literature review have implications for future research and practice in the field of brain tumor segmentation.

The current state of the field suggests that there is still room for improvement, particularly in terms of increasing the accuracy and robustness of the models. Future research may focus on incorporating more advanced techniques, such as graph convolutional networks, adversarial training, and transfer learning, to further enhance the performance of brain tumor segmentation models. Additionally, the use of large, diverse, and well-annotated medical imaging datasets will be crucial for training and validating these models.

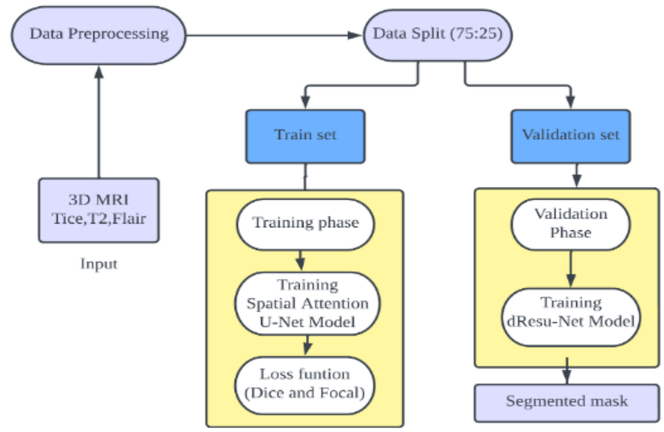

Fig. 1. The Proposed Methodology's General Flow

III. Material & Methods

In this paper, we have used a Brats2020 dataset for training a Spatial Attention Modified U-Net model with attention mechanisms for the segmentation of brain tumors in multi-modalities MRI images. The general workflow of our methodology is shown in Figure 1 and sample images of brain tumors in different modalities (flair, T1, T1CE, T2) along with tumor mask are shown in Figure 2.

U-Net-based models are widely used for the image segmentation task and it has an encoder and decoder for performing this operation. The Encoded part takes the 3D volume as input of size $128 \times 128 \times 128 \times 3$. There are a total of 4 layers in the model including the input layer. Channel-wise attention mechanisms are applied at different levels of the model during encoding and decoding which help in avoiding feature loss. From the top to the bottom of the encoder, the number of feature maps increases, and the size of feature maps reduces. In the decoder, the dimension of the feature map is doubled after each upsampling. The top of the decoder is the output layer consisting of a convolution layer and a sigmoid activation function. Further, testing is performed on the Brats 2021 dataset on 50% of the dataset. The architectural model of our approach is shown in Figures 3, 4, and 5.

Algorithm 1 LoadImages

function LoadImages(*img_dir*, *img_list*)
  images ← empty list
  for i, *image_name* **in enumerate** *img_list* **do**
    if $(image\_name.split('.')[1] ==' npy')$ **then**
      image ← load image from *img_dir+image_name*
      images.append(image)
    end if
  end for
  images ← numpy array of *images*
  return *images*
end function=0

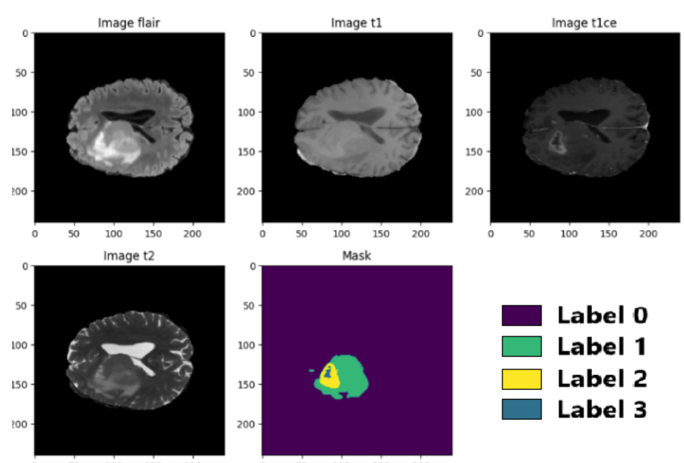

Fig. 2. Sample image of multi-modality MRI scans of Brain Dataset 2020.

IV. Pre-processing

A custom generator is generated and preprocessing is done for the BraTS2020 dataset. The BraTS2020 dataset consists of MRI scans of brain tumors and their corresponding segmentation masks. Steps:

- Define a MinMaxScaler object that scales the intensity values of the MRI scans.
- The scaler is used to normalize the data to a range of 0 to 1 to improve the convergence of the deep learning model.

Algorithm 2 ImageLoader

```
function IMAGELOADER(img_dir, img_list, mask_dir, mask_list, batch_size)
    L ← length of img_list
    while True do
        batch_start ← 0
        batch_end ← batch_size
        while batch_start < L do
            limit ← min(batch_end, L)
            X ← LoadImages(img_dir, img_list[batch_start : limit])
            Y ← LoadImages(mask_dir, mask_list[batch_start : limit])
            yield (X, Y)
            batch_start ← batch_start + batch_size
            batch_end ← batch_end + batch_size
        end while
    end while
end function=0
```

- The MRI scans are stored in separate lists for each imaging modality, i.e., T2, T1ce, and FLAIR, while the masks are stored in a separate list.
- Load the MRI scans and their corresponding masks using the nibabel library.
- Scale the intensity values of the MRI scans using the MinMaxScaler object
- Stack the three MRI scans into a single 3D image volume.
- Crop the image volume and its corresponding mask to remove any empty borders.
- Convert the mask to a one-hot encoded representation using the to_categorical function from tensorflowk̇erasu̇tils.
- Save the preprocessed image and mask as NumPy arrays.
- Then the preprocessed data is split into training and validation sets with a ratio of 75:25.

V. Implementation Details and Results

In this section, we have mentioned details of various loss functions used during the training and validation phase. The details of hyperparameters and results obtained on testing sets are also mentioned.

A. Loss Function

Dice Loss Function: It finds the similarity between two samples and is represented as:

$$DiceLoss = 1 - (2 * TP)/(2 * TP + FP + FN) \quad (1)$$

where TP (True Positive) is the number of pixels that are correctly classified as positive, FP (False Positive) is the number of pixels that are incorrectly classified as positive, and FN (False Negative) is the number of pixels that are incorrectly classified as negative.

Algorithm 3 Model Training

```
procedure TRAINMODEL
    Define loss as loss
    Define metrics as metrics
    Define optimizer as optimizer
    Set batch_size
    Set steps_per_epoch
    Build 3D U-Net model with attention
    Train the model for specified epochs
    for each epoch in range(num_epochs) do
        for each step in range(steps_per_epoch) do
            Load batch of images and masks from train_gen
            Perform forward and backward passes
            Update model parameters
            Record loss and metrics
        end for
        Load saved model
        Initialize mIOU as 0
        for each batch of validation images and masks in val_gen do
            Calculate mIOU for the batch
            Predict masks using loaded model
            Calculate mIOU between predicted masks and ground truth masks
        end for
        Select an image for prediction
        Load and preprocess test image
        Predict mask for the test image
    end for
end procedure=0
```

B. Categorical Focal Loss

A Focal Loss function addresses class imbalance during training in tasks like object detection.

$$FocalLoss = -\alpha(1 - p)^{\gamma} \log(p) \quad (2)$$

where α is the balancing parameter, p is the predicted probability, and is the focusing parameter. The balancing parameter is used to balance the contribution of each class to the loss function, while the focusing parameter is used to control the rate at which the loss function focuses on hard-to-classify samples.

C. Metrics

The total loss function is a combination of the Dice Loss and the Categorical Focal Loss. The weights of the two loss

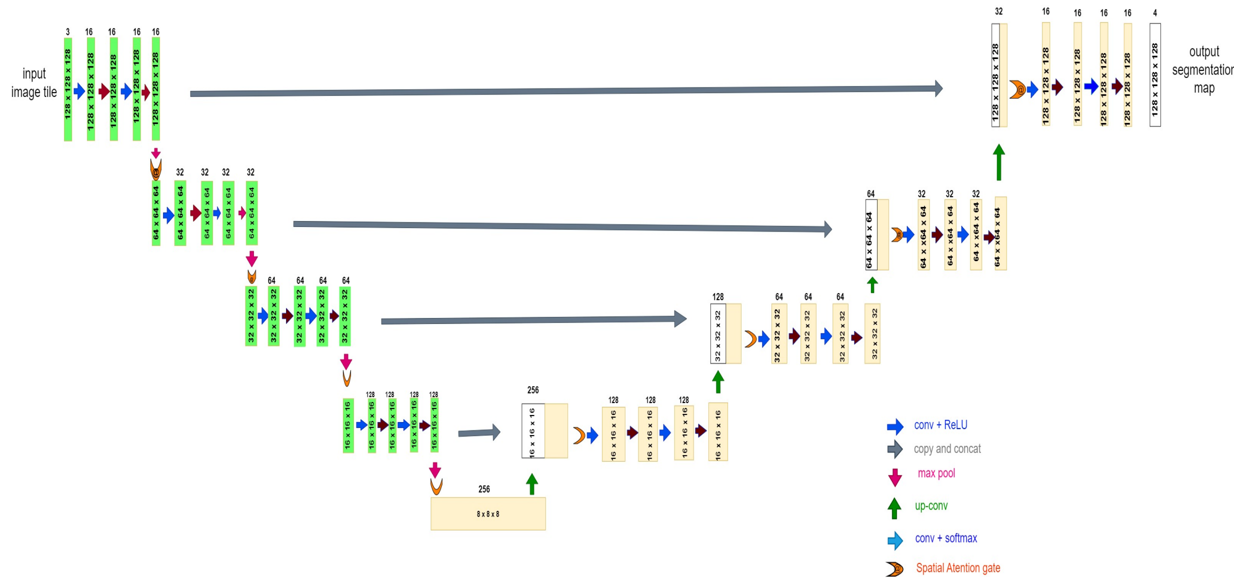

Fig. 3. Spatial attention Based Architecture

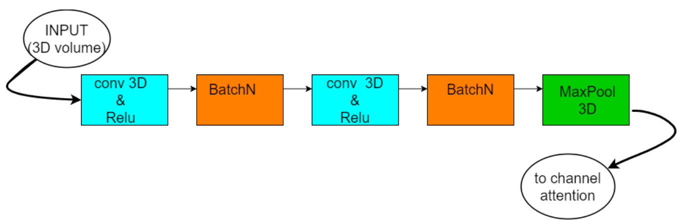

Fig. 4. Operations of Each layer of Contracting path

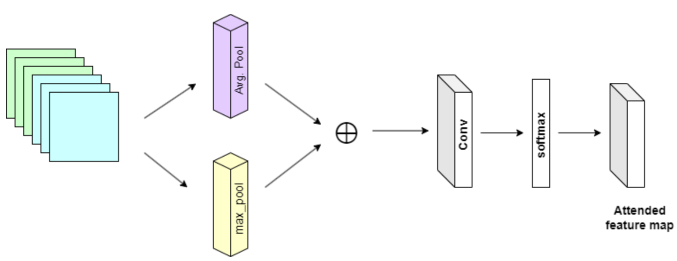

Fig. 5. Spatial Attention Module Overview

functions are adjusted using a hyperparameter to control the trade-off between them during the training process. In the stated paper, the weight of the Categorical Focal Loss is set to 1.

The formula for Intersection over Union (IoU) Score is as follows:

$$IoUScore = TP/(TP + FP + FN) \tag{3}$$

where TP (True Positive) is the number of pixels that are correctly classified as part of the object of interest, FP (False Positive) is the number of pixels that are incorrectly classified as part of the object of interest, and FN (False Negative) is the number of pixels that are incorrectly classified as not part of the object of interest.

D. Optimiser - Adam optimiser

The Adam optimizer is a popular optimization algorithm for training neural networks. It is a stochastic gradient descent method that uses adaptive learning rates to update the weights of the neural network. The algorithm combines the advantages of two other popular optimization algorithms, Adagrad and RMSprop. The Adam optimizer uses two momentum parameters, beta1 and beta2, to control the decay rate of the moving averages of the gradient and the squared gradient, respectively. The update rule for the weights is given by [**?**]:

$$w(t+1) = w(t) - \eta \frac{m(t)}{\sqrt{v(t)} + \epsilon} \tag{4}$$

where w(t) is the weight at time step t, is the learning rate, m(t) is the moving average of the gradient, v(t) is the moving average of the squared gradient, and epsilon is a small value added to the denominator to prevent division by zero.

In this section, the details about the evaluation measures employed to assess the performance of the proposed model, implementation details, and results achieved from the proposed method are discussed.

E. Evaluation Measure

- Dice coefficient score (DSC), specificity, and sensitivity are used to assess the proposed approach

$$Dice\_coefficient = 2TP/2TP + FP + FN \tag{5}$$

- A binary cross-entropy loss function has been used.

F. Implementation Detail

- The proposed model (dResU-Net with spatial attention) was implemented using Python programming language, Keras library, and TensorFlow as the backend. The proposed model was implemented using Python programming language, Keras library, and TensorFlow as the backend.
- For the experimental purpose, an ADAM optimizer with a learning rate of 0.0001 was used. The activation function ReLU with batch normalization was employed. Batch normalization normally increases the stability of the model and normalizes the network at each layer.
- The model was trained for 100 epochs on a batch size of 2 due to the limited computational resources. The experiments were conducted on the BraTS 2020 benchmark dataset, from which 75% of the data was used for training and 25% data for validation.
- The testing was done on 50% of the Brats 2021 benchmark dataset, it was tested for the batch size of 6.

TABLE I
HYPERPAREMETERS OF PROPOSED SPATIAL ATTENTION MODEL

Hyperparameters	**values**
Input size	256 × 256 × 256 ×3
Learning rate	0.0001
batch Size	2
The hidden layer activation function	ReLU
Optimizer	ADAM
Loss function	binary cross-entropy loss function
No. of epochs	100
Dropout	0.1 - 0.2
Output size	128 × 128 × 128 × 3
Output layer activation function	Softmax

G. Analysis of the results

1) Training and Validation Details: The training loss per epoch and validation loss per epoch are shown with The Accuracy Per epoch of training and validation.

- The Training and validation were done on the Brats2020 dataset which consists of 369 images
- After preprocessing we got 344 useful images.
- These 344 further split into 75:25 ratio for training and validation that is 258 for training and 86 for validation

2) Testing details:

- The Testing was done on the Brats2021 dataset which consists of 1250 images which took 50% of the images that is 625 Images.
- After preprocessing we got 576 useful images

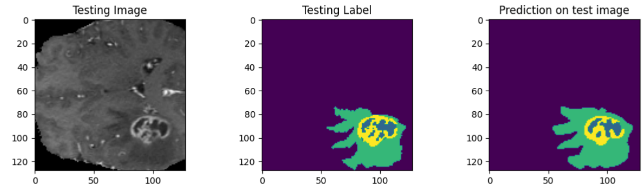

Fig. 6. Simple UNet Architecture Prediction Image

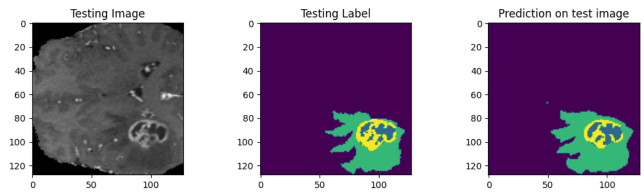

Fig. 7. Modified UNet Architecture Prediction Image

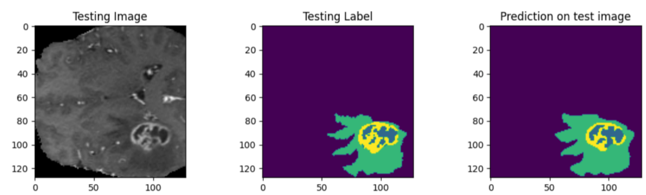

Fig. 8. Spatial Attention UNet Architecture Prediction Image

VI. CONCLUSION

Paper introduces an innovative approach to brain tumor segmentation using a modified 3D U-net with spatial attention. Accurate segmentation is crucial for diagnosis and treatment planning. Our model's channel attention mechanism effectively preserves local details during encoding and decoding, leading to impressive segmentation results on the BraTS 2020 dataset (dice scores of 0.9027, 0.8868, and 0.9067 for TC, WT, and ET). The model maintains its robustness on an external cohort from BraTS 2021 (dice scores of 0.8039, 0.7640, and 0.6689). This work holds promise for advancing brain tumor diagnosis and treatment by combining U-net's power with strategic attention mechanisms, paving the way for further research in the field.

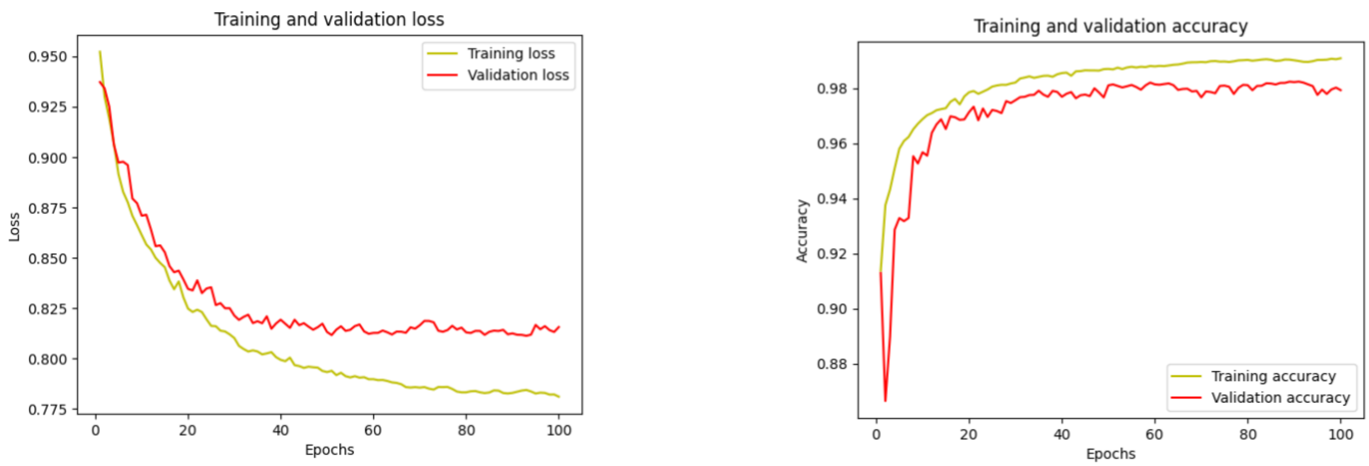

Fig. 9. Simple Unet Training And Validation Loss with Training and Validation accuracy

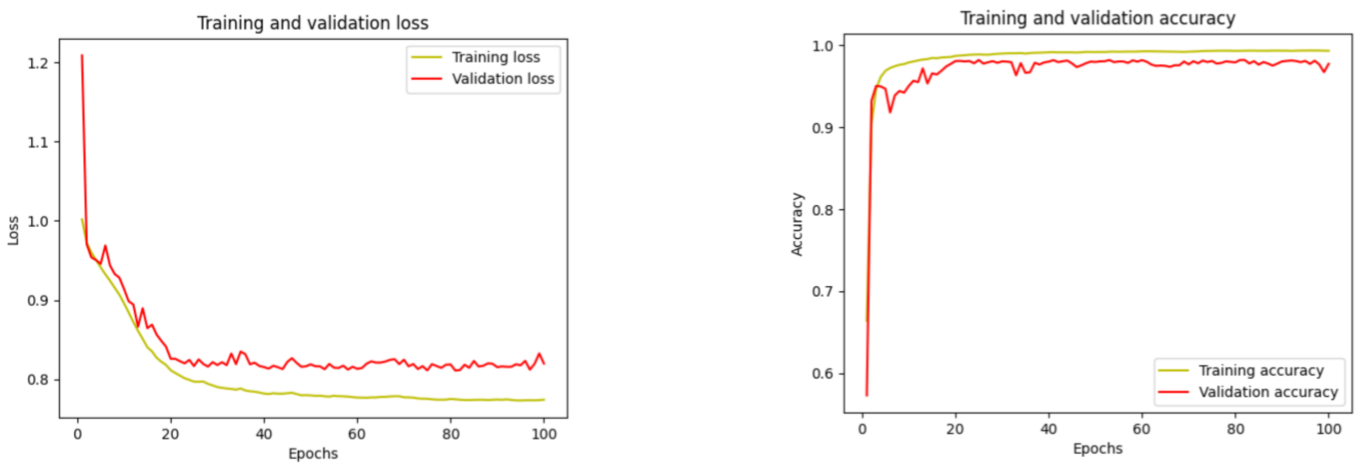

Fig. 10. Modified Unet Training And Validation Loss with Training and Validation accuracy

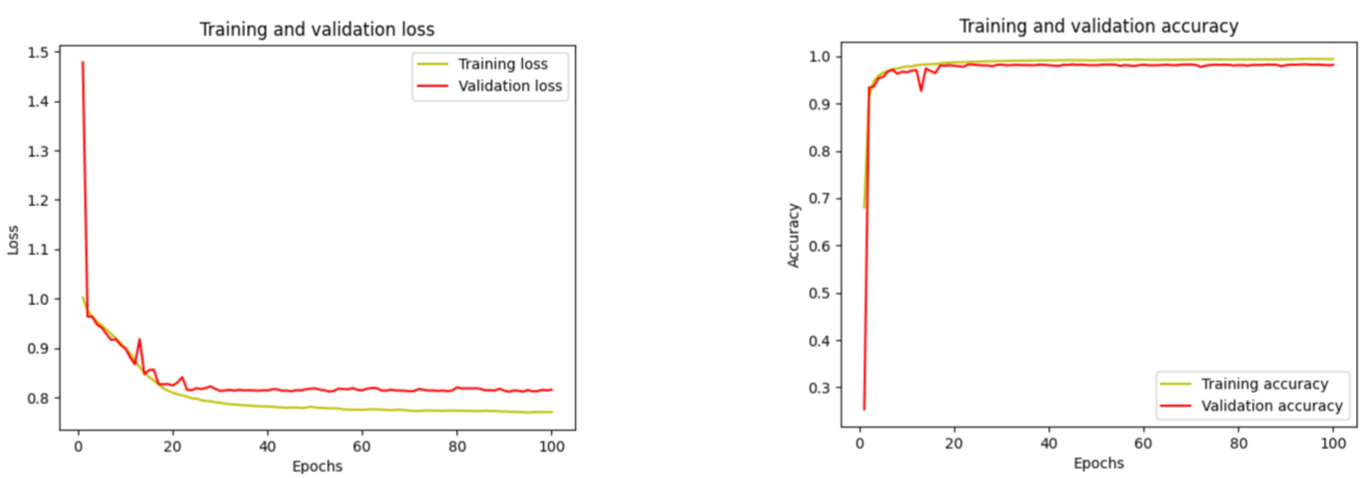

Fig. 11. Spatial attention Unet Training And Validation Loss with Training and Validation accuracy

REFERENCES

[1] Miller, KD, Ostrom, QT, Kruchko, C, Patil, N, Tihan, T, Cioffi, G, Fuchs, HE, Waite, KA, Jemal, A, Siegel, RL, Barnholtz-Sloan, JS. Brain and other central nervous system tumor statistics, 2021. CA Cancer J Clin. 2021. https://doi.org/10.3322/caac.21693

[2] Holland EC. Progenitor cells and glioma formation. Curr Opin Neurol. 2001 Dec;14(6):683-8. doi: 10.1097/00019052-200112000-00002. PMID: 11723374.

[3] Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P., & Larochelle, H. (2017). Brain tumor segmentation with Deep Neural Networks. Medical Image Analysis, 35, 18-31. https://doi.org/10.1016/j.media.2016.05.004

[4] Montaha, S., Azam, S., Rafid, A. K. M. R. H., Hasan, M. Z., Karim, A. (2023). Brain Tumor Segmentation from 3D MRI Scans Using U-Net. SN Computer Science, 4(4). https://doi.org/10.1007/s42979-023-01854-6

[5] Raza, R., Bajwa, U. I., Mehmood, Y., Anwar, M. W., Jamal, M. (2023). dResU-Net: 3D deep residual U-Net based brain tumor segmentation from multimodal MRI. Biomedical Signal Processing and Control, 79, 103861. https://doi.org/10.1016/j.bspc.2022.103861

[6] Wang, J., Lv, P., Wang, H., Shi, C. (2021b). SAR-U-Net: Squeeze-and-excitation block and atrous spatial pyramid pooling based residual U-Net for automatic liver segmentation in Computed Tomography. Computer Methods and Programs in Biomedicine, 208, 106268. https://doi.org/10.1016/j.cmpb.2021.106268

[7] Wang, W. (2021b, March 7). TransBTS: Multimodal brain tumor segmentation using transformer. arXiv.org. https://arxiv.org/abs/2103.04430

[8] Spatial attention-based efficiently features fusion network for 3D-MR brain tumor segmentation. (2020, December 18). IEEE Conference Publication — IEEE Xplore. https://ieeexplore.ieee.org/document/9350767

[9] Ladkat, A. S., Bangare, S. L., Jagota, V., Sanober, S., Beram, S. M., Rane, K. P., Singh, B. K. (2022). Deep Neural Network-Based Novel Mathematical Model for 3D Brain Tumor Segmentation. Computational Intelligence and Neuroscience, 2022, 1–8. https://doi.org/10.1155/2022/4271711

[10] Hypercolumn technique. (n.d.). https://bio-protocol.org/exchange/minidetail?type=30id=9153745