UTILIZING ATTENTION, LINKED BLOCKS, AND PYRAMID POOLING TO PROPEL BRAIN TUMOR SEGMENTATION IN 3D

Anonymous authors
Paper under double-blind review

ABSTRACT

We present an approach to detect and segment tumorous regions of the brain by establishing three varied segmentation architectures for multiclass semantic segmentation along with data specific customizations like residual blocks, soft attention mechanism, pyramid pooling, linked architecture and 3D compatibility to work with 3D brain MRI images. The proposed segmentation architectures namely, Attention Residual UNET 3D also referred to as AR-UNET 3D, LinkNet 3D and PSPNet 3D, segment the MRI images and succeed in isolating three classes of tumors. By assigning pixel probabilities, each of these models differentiates between pixels belonging to tumorous and non-tumorous regions of the brain. By experimenting and observing the performance of each of the three architectures using metrics like Dice loss and Dice score, on the BraTS2020 dataset, we successfully establish quality results.

1 INTRODUCTION

Over the years, manual segmentation of brain tumors has turned out to be tedious for the radiologists and doctors alike due to the amount of time and precision it requires to identify such tumors. Hence, making use of an automated system employing Deep Learning and Computer Vision will reduce the operator fatigue and manual errors.

Brain tumor segmentation involves identifying the tumorous region in the brain (pixels indicating the tumorous cells in the MRI scans). Hence, for the purpose of segmenting brain tumors, we present three novel 3D architectures.

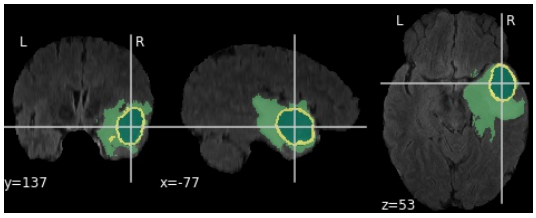

Figure 1: View of the coronal, sagittal and axial planes of the brain

The Attention Residual UNET 3D or AR-UNET 3D is a modification upon the existing Residual U-Net (Zhang et al., 2017) and Attention U-Net (Oktay et al., 2018), both of which operate in 2D. AR-UNET 3D makes use of the residual blocks from ResNet (He et al., 2016) which help in maintaining skip connections using identity mappings while the proposed soft attention mechanism provides an added advantage by weighing the more important features, heavily.

The LinkNet 3D is a modification upon the existing LinkNet (Chaurasia & Culurciello, 2017) in 2D. LinkNet 3D makes use of residual blocks from ResNet18 3D (He et al., 2016) in its encoder for feature extraction, and links the output from each encoder block to its corresponding decoder block to account for lost spatial information due to multiple downsampling.

The PSPNet 3D is a modification upon the existing PSPNet (Zhao et al., 2017) in 2D. PSPNet 3D makes use of a 3D Pyramid Pooling Module for interpolating the 3D feature maps into different resolutions that facilitates efficient extraction of spatial information and global context capturing. This helps the model learn about the spatial context associated with different classes of the image.

In the field of 3D MRI Brain Tumor Segmentation, many architectures like 3D UNET (Özgün Çiçek et al., 2016), VNET (Milletari et al., 2016), AutoEncoder (Bank et al., 2020), etc. have been used to achieve state-of-the-art results. We propose 3D advancements of established models which are well-known for their segmentation tasks in 2D.

2 Related works

Global context capturing in 2D has always been an important aspect of semantic segmentation. Various deep learning techniques such as dynamic context-aware kernels (He et al. (2019)), attention mechanisms (Zhao et al. (2018)) and Laplacian Pyramids (Ghiasi et al. (2016)) have been adopted in the past to explore this aspect of semantic segmentation. Tumor segmentation using Deep Learning by Type Specific Sorting of Images (Sobhaninia et al. (2018)) was trained separately for sagittal, axial and coronal views by using LinkNet (Chaurasia & Culurciello (2017)). All these aforementioned methods have shown impressive results for 2D image segmentation while there has been a need of extensive research to incorporate the same for 3D images.

Some of the 3D architectures like the 3D AGSE-VNet (Guan et al. (2021)), nnU-Net (Isensee et al. (2020)), KiU-Net (Valanarasu et al. (2020)), etc. have achieved state-of-the-art results in 3D segmentation. The 3D AGSE-VNet (Guan et al. (2021)) uses Squeeze and Excite (SE) module (Sultana et al. (2018)) in each of its encoder and Attention Guided Filter (Zhang et al. (2019)) in each of its decoder and has achieved a Dice score (Sudre et al. (2017)) of 0.68 for the whole tumor. The nnU-Net (Isensee et al. (2020)) makes use of a 3D U-Net (Özgün Çiçek et al. (2016)) as its baseline and normalized input images to perform segmentation and it is further modified by updating certain aspects like region based training, batch size, augmentation techniques and more. This model achieved mean Dice score (Sudre et al. (2017)) of 0.8529. We propose this paper to establish results for brain tumor segmentation in 3D.

3 Method

3.1 Attention Residual UNET (AR-UNET) 3D:

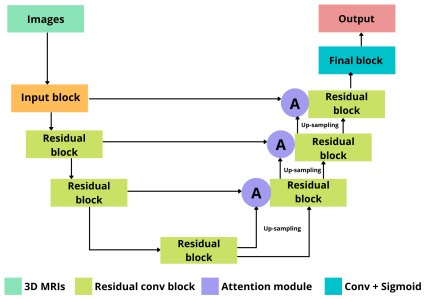

Figure 2: AR-UNET 3D

The AR-UNET 3D, as depicted in Figure 2 and described in Table 1 is a 3D advancement of the Residual U-Net (Zhang et al., 2017) and Attention U-Net (Oktay et al., 2018) models that segment in 2D. The AR-UNET 3D makes use of residual blocks based on the ResNet (He et al., 2016) architecture and possesses the added advantage of soft attention mechanism. This attention mechanism helps the model learn more important features from previous layers by assigning weights that cancel out less relevant ones while maintaining the importance of features that can help the model learn better.

The encoder passes a given image through convolutional and residual blocks for downsampling it. The image ends up with 256 channels after a pass through the encoder. The bridge, which is a residual block, connects the encoder module to the decoder module. The decoder consists of upsampling layers through which the image is passed before being concatenated with respective encoder outputs. Weighted matrices obtained from the attention module are also multiplied to the features to help the model determine which important features need to be considered while learning the weights.

Further, a final output block produces the segmented image which consists of the 4 channel mask that represents the foreground (tumorous) region with 3 classes and the background region with the no-tumor class.

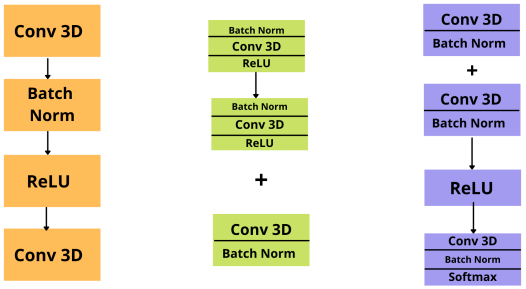

Figure 3: Input, residual and attention blocks in AR-UNET 3D

Residual block: The second block in Figure 3 consists of (3x3x3) convolutional layers with skip connections that aid in retaining important features by using identity mappings and learning the residuals similar to the ResNet (He et al., 2016) architecture which uses such residual blocks. Each block consists of two modules as can be seen Figure 3 and a final skip connection which is summed to the former combination.

Attention block: The third block in Figure 3 takes in 2 inputs; one from the encoder and another from the previous layer's decoder. Both are passed through (1x1x1) convolutions and are further summed up. This combination is passed through another convolutional module which produces a 1 channel output or a singular weight matrix. This aids in obtaining a combined weight value which is then multiplied to the original input. Such a process of calculating weight and implementing soft attention guarantees a higher level of importance to relevant features.

Final block: Further, the obtained output is passed through a (1x1x1) convolutional layer and softmax activation function. Thus, the output is condensed into the desired result which represents a segmented mask with pixel probabilities.

Table 1: AR-UNET 3D - Modules and blocks

Modules	Blocks	Action	Out channels
Encoder	Initial input block + skip	Input block with skip connection	64
Bridge	Residual block 1 and 2 Bridge residual block	1st and 2nd block in the Encoder Connects Encoder to Decoder using residual block	128,256 512
Decoder x 3	Upsample Attention blocks Residual blocks	Upsamples the image Weighs inputs Converts concatenated channels to required size	512, 256, 128 512, 256, 128 256, 128, 64
Final	Final output block	Produces a segmented mask	4

3.2 LINKNET 3D:

The proposed architecture, as can be seen in Figure 4 and described in Table 2 is a 3D modification upon the LinkNet architecture (Chaurasia & Culurciello, 2017) for semantic segmentation in 2D space. The 'conv' here refers to a convolution operation in 3D space. Batch Normalization (Ioffe & Szegedy, 2015) is used between each convolutional layer, followed by ReLU (Nair & Hinton, 2010) activation to introduce non-linearity as has been mentioned in (Chaurasia & Culurciello, 2017). The LinkNet 3D network comprises of an Encoder network which is effectively a 3D variation of the 18 layer Residual Network or ResNet18 (He et al., 2016), and a Decoder network which is a modified 3D variation of the decoder architecture mentioned in (Chaurasia & Culurciello, 2017).

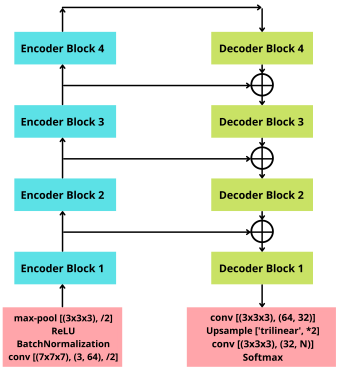

Figure 4: LinkNet 3D

Encoder: The encoder network comprises an initial block and 4 residual blocks (He et al., 2016) for feature extraction purposes.

Initial Block: The initial block of the encoder performs convolution operation on the input image, with a (7x7x7) kernel and a stride of 2. This is followed by Batch Normalization (Ioffe & Szegedy, 2015) and ReLU activation (Nair & Hinton, 2010) before a spatial max-pooling operation with a kernel size of (3x3x3) and a stride of 2.

Residual Blocks: Following the initial block of the encoder are the residual blocks (He et al., 2016), used for feature extraction. They are represented in Figure 4 as Encoder Block (i).

Each layer within the residual blocks is shown in Figure 5 in detail. Each residual block has a strided convolution operation followed by 3 convolutional operations with a (3x3x3) kernel accompanied by skip connections (Orhan & Pitkow, 2017).

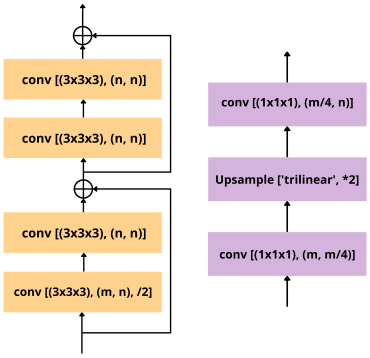

Figure 5: Each Encoder(left) and Decoder(right) block (convolutional modules)

Linked Architecture: The linking of each encoder to each decoder has been performed exactly as mentioned in (Chaurasia & Culurciello, 2017), in order to recover the lost spatial information which is lost due to multiple downsampling operations in the encoder. To enable the linking operation, strided convolutions are used in the encoder.

Decoder: The decoder network takes the output from the encoder as its input. It comprises 4 decoder blocks and a final segmentation block for performing the main segmentation task.

Decoder Blocks: The decoder blocks are represented as Decoder Block (i) in Figure 4. Each layer within the decoder block is shown in Figure 5 in detail. Each decoder block has 2 convolutional operations with a (1x1x1) kernel and an Upsample operation in trilinear mode with scale-factor 2, between them. The decoder blocks are followed by the final segmentation block.

Final Segmentation Block: The final segmentation block of the decoder performs the main segmentation task on the output received from the Decoder Block 1. A convolution operation with kernel size (3x3x3) is performed on the decoder output, followed by an Upsample operation with scale-factor 2 in trilinear mode. Finally, another convolution operation with kernel size (3x3x3) is performed before passing it through a softmax layer to get the final segmentation mask with pixel probabilities.

Table 2: LinkNet 3D - Modules and blocks

Modules	Blocks	Action	Out channels
	Initial Input block	Convolution and max-pooling	64
Encoder	Residual blocks	4 blocks following the initial input block for 'Feature extraction'	64, 128, 256, 512
Decoder	Decoder Blocks	Converts concatenated channels (512, 256, 128, 64) to required channel size	256, 128, 64, 64
	Final Block	Produces the final segmentation mask	4

3.3 Pyramid Scene Parsing Network (PSPNet) 3D:

The proposed architecture as can be seen in Table 3, namely PSPNet-3D is a three-dimensional adaptation of the original PSPNet paper (Zhao et al., 2017) which was used for 2D segmentation. PSPNet-3D is specialized to work on 3 dimensional images by utilizing the techniques of global context aggregation and local level predictions.

The methodology follows extracting spatial features from a 3D encoder network which uses dilated convolutions (Yu & Koltun, 2016) followed by a decoder network that performs the required semantic segmentation.

3D Dilated Residual Blocks: These act as the feature extractor backbones which use dilated convolutions (Yu & Koltun, 2016) with wider receptive fields that allow enhanced extraction of spatial information. Here, the dilated Resnet50-3D (Hara et al., 2017) which is a modification of the original ResNet50 (He et al., 2016), serves as the encoder network with dilations of 2 and 4 introduced in the last two residual blocks of the network. The output from the initial input block is passed sequentially through 4 residual blocks to generate it's feature representation. Figure 6 below represents the residual blocks of the encoder along with their respective kernel sizes, output dimensions, stride and dilations.

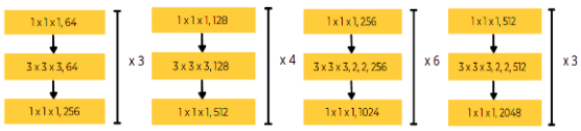

Figure 6: Residual blocks 1, 2, 3 and 4

Decoder: The decoder network comprises of the 3D Pyramid Pooling Module and the Final Segmentation Block

3D Pyramid Pooling Module: The 3D pyramid pooling module is at the core of the PSPNet-3D architecture. The feature map from the final residual block of the encoder is passed into the 3D pyramid pooling network which then uses four 3D adaptive average pooling layers for feature pooling at four different resolutions (1, 2, 3 and 6). This allows the model to learn the context of the overall image at four different levels while retaining the spatial information of the image. The 3D pyramid pooling blocks are demonstrated in Figure 7.

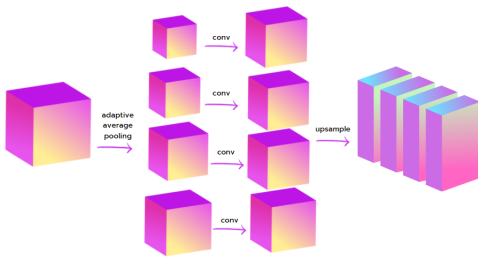

Figure 7: 3D Pyramid Pooling Module

The pooled feature maps are then upsampled through a convolution layer and are concatenated with the original feature map before passing it to the final segmentation block.

Final Segmentation Block: Three successive convolution operations are performed on the concatenated feature block with a 1x1x1 filter. This helps the network to interpret the encoded contextual information of the image previously captured at different levels. The generated output is then passed through a softmax layer to produce the final mask.

The complete architecture is a sequential combination of all the aforementioned sub-parts. The encoder and the decoder are jointly used to efficiently extract both contextual and spatial information from 3D images and perform segmentation on the same. Table 3 depicts the complete model architecture.

Table 3: PSPNet 3D - Modules and blocks

Modules	Blocks	Action	Out channels
	Initial Input block	3 convolutions and a max-pooling block.	128
Encoder	3D Dilated Residual blocks	4 residual blocks for feature extraction following the initial input block	256, 512, 1024, 2048
Decoder	3D Pyramid Pooling Module	4 pooling blocks of 4 different reolutions generating the final concatenated feature map.	1024
	Final Block	Produces the final segmentation mask	4

4 EXPERIMENTS

4.1 DATASET

The (Brain Tumor Segmentation) BraTS 2020 dataset (Menze et al., 2015) (Bakas et al., 2017) (Bakas et al., 2018) with a total of 250 brain MRI 3D images belonging to flair modality, in which

eight percent of the training data has been substituted with 3D augmentations (Nalepa et al., 2019) based on time tested augmentation techniques like affine transforms and flips for each of the respective models.

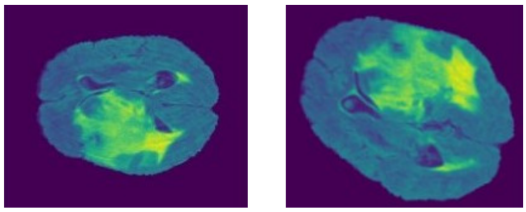

Figure 8: Flipped and affine transformed images

Pre-processing techniques like resize, normalization and standardization have been performed along with augmentations like 3D flip and affine transformations. All images with 64 slices were processed to contain zero mean and unit standard deviation as a part of the normalization procedure.

The normalized images were first flipped along the lateral axis followed by affine transformation applied over them which included re-scaling, rotation and linear interpolation over the pixel intensities.

Further, the masks were processed to contain one hot encoded representations of four classes of tumors; 0: no tumor, 1: non enhancing tumor core, 2: edema and 3: enhancing tumor, thereby making it a multiclass segmentation task.

4.2 METRICS

4.2.1 DICE SCORE AND LOSS

We have adopted the Dice Loss (Sudre et al., 2017) as the primary evaluation metric for our segmentation task. We consider the ground truth and the predicted label as the two vectors. The dice coefficient was computed using the following formula -

$$Dice\ coefficient\ =\ \frac{2\,|T \cap P|}{|T| + |P|}$$

Here, T and P denote the vectors corresponding to the true and the predicted classes respectively.

The generalized loss function is formulated from the dice score by subtracting the dice coefficient from 1. The dice loss is minimized to simultaneously maximize the dice coefficient since a higher dice coefficient implies a better overlap.

$$Dice\ Loss\ =\ 1 - Dice\ coefficient$$

4.3 EXPERIMENT RESULTS

4.3.1 TRAINING

All 3 segmentation models were trained using the PyTorch (Paszke et al., 2019) framework on the Tesla P100 PCIe GPU with 250 images of batch size 4, for 200 iterations using the RMSProp optimization algorithm.

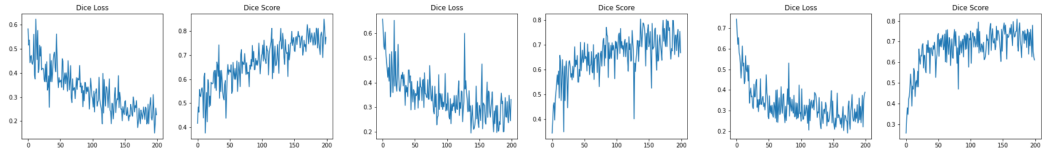

Figure 9: Training results from AR-UNET 3D, LinkNet 3D and PSPNet 3D (left to right)

Table 4: Training details

Models	Parameters	Learning Rate	Mean Dice Loss	Best Dice Score
AR-UNET 3D	35838349	0.0001	0.3270	0.8500
LinkNet 3D	32925860	0.00001	0.3431	0.8041
PSPNet 3D	74352202	0.0001	0.3334	0.8087

Test results for all 3 models are given below.

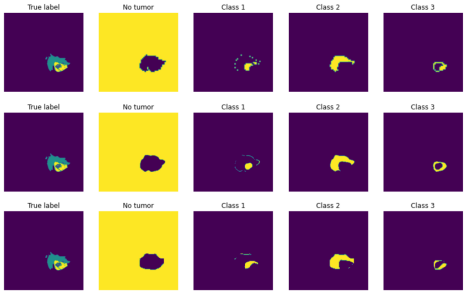

Figure 10: Test results from AR-UNET 3D, LinkNet 3D and PSPNet 3D (top to bottom)

5 Conclusion and Future Works

Due to computational shortcomings, each model was trained on a restricted amount of data. Exposing the models to higher level of augmentations, more slices and full volumes will help the models segment even better. We recommend using higher configuration GPUs and stable training environments for achieving full capacity of these models.

With available resources and computational power, the successful segmentation of 3D MRI images has been performed using 3 distinct architectures namely, AR-UNET 3D, LinkNet 3D and PSPNet 3D. The results have been inferred along with test results consisting of accurate segmentation by each of the three models.

References

Spyridon Bakas, Hamed Akbari, Aristeidis Sotiras, Michel Bilello, Martin Rozycki, Justin S Kirby, John B Freymann, Keyvan Farahani, and Christos Davatzikos. Advancing the cancer genome atlas glioma mri collections with expert segmentation labels and radiomic features. *Scientific data*, 4 (1):1–13, 2017.

Spyridon Bakas, Mauricio Reyes, Andras Jakab, Stefan Bauer, Markus Rempfler, Alessandro Crimi, Russell Takeshi Shinohara, Christoph Berger, Sung Min Ha, Martin Rozycki, et al. Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge. *arXiv preprint arXiv:1811.02629*, 2018.

Dor Bank et al. Autoencoders. 2020.

Abhishek Chaurasia and Eugenio Culurciello. Linknet: Exploiting encoder representations for efficient semantic segmentation. pp. 1–4, 2017.

Golnaz Ghiasi et al. Laplacian pyramid reconstruction and refinement for semantic segmentation. *Computer Vision – ECCV*, 9907, 2016.

Xi Guan et al. 3d agse-vnet: An automatic brain tumor mri data segmentation framework. 2021.

Kensho Hara et al. Learning spatio-temporal features with 3d residual networks for action recognition. pp. 3154–3160, 2017.

Junjun He et al. Dynamic multi-scale filters for semantic segmentation. pp. 3561–3571, 2019.

Kaiming He et al. Deep residual learning for image recognition. 2016.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. 2015.

Fabian Isensee et al. nnu-net for brain tumor segmentation. *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*, 12659, 2020.

Bjoern H. Menze et al. The multimodal brain tumor image segmentation benchmark (brats). *IEEE Transactions on Medical Imaging*, 34(10):1993–2024, 2015. doi: 10.1109/TMI.2014.2377694.

Fausto Milletari et al. V-net: Fully convolutional neural networks for volumetric medical image segmentation. pp. 565–571, 2016.

Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. *Proceedings of ICML*, 27:807–814, 2010.

Jakub Nalepa et al. Data augmentation for brain-tumor segmentation: A review. *Frontiers in Computational Neuroscience*, 13, 2019.

Ozan Oktay et al. Attention u-net: Learning where to look for the pancreas. 2018.

A. Emin Orhan and Xaq Pitkow. Skip connections eliminate singularities. 2017.

Adam Paszke et al. Pytorch: An imperative style, high-performance deep learning library. 2019.

Zahra Sobhaninia et al. Brain tumor segmentation using deep learning by type specific sorting of images. 2018.

Carole H Sudre et al. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. 10553, 2017.

Farhana Sultana et al. Advancements in image classification using convolutional neural network. pp. 122–129, 2018.

Jeya Maria Jose Valanarasu et al. Kiu-net: Overcomplete convolutional architectures for biomedical image and volumetric segmentation. 2020.

Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. 2016.

Shihao Zhang et al. Attention guided network for retinal image segmentation. 2019.

Zhengxin Zhang et al. Road extraction by deep residual u-net. *IEEE GEOSCIENCE AND REMOTE SENSING LETTERS*, 2017.

Hengshuang Zhao et al. Pyramid scene parsing network. pp. 6230–6239, 2017.

Hengshuang Zhao et al. Psanet: Point-wise spatial attention network for scene parsing. *Computer Vision – ECCV 2018*, 11213, 2018.

Özgün Çiçek et al. 3d u-net: Learning dense volumetric segmentation from sparse annotation. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, 9901, 2016.