A multi-path decoder network for brain tumor segmentation

Yunzhe Xue[1], Meiyan Xie[1], Fadi G. Farhat[1], Olga Boukrina[2], A.M. Barrett[3], Jeffrey R. Binder[4], Usman W. Roshan[1], and William W. Graves[5]

[1] Department of Computer Science, New Jersey Institute of Technology, Newark NJ, USA
[2] Stroke Rehabilitation Research, Kessler Foundation, West Orange, NJ, USA
[3] Emory University and Atlanta VA Medical Center, Atlanta, GA, USA
[4] Department of Neurology, Medical College of Wisconsin, Milwaukee, WI, USA
[5] Department of Psychology, Rutgers University – Newark, Newark, NJ, USA

Abstract. The identification of brain tumor type, shape, and size from MRI images plays an important role in glioma diagnosis and treatment. Manually identifying the tumor is time expensive and prone to error. And while information from different image modalities may help in principle, using these modalities for manual tumor segmentation may be even more time consuming. Convolutional U-Net architectures with encoders and decoders are state of the art in automated methods for image segmentation. Often only a single encoder and decoder is used, where different modalities and regions of the tumor share the same model parameters. This may lead to incorrect segmentations. We propose a convolutional U-Net that has separate, independent encoders for each image modality. The outputs from each encoder are concatenated and given to separate fusion and decoder blocks for each region of the tumor. The features from each decoder block are then calibrated in a final feature fusion block, after which the model gives it final predictions. Our network is an end-to-end model that simplifies training and reproducibility. On the BraTS 2019 validation dataset our model achieves average Dice values of 0.75, 0.90, and 0.83 for the enhancing tumor, whole tumor, and tumor core subregions respectively.

Keywords: Convolutional neural networks · multi-modal · brain MRI

1 Introduction

Gliomas are the most commonly occurring tumor in the human central nervous system [1]. They fall into low-grade (LGG) and high-grade (HGG) subtypes and have three subregions: the enhanced tumor, tumor core, and whole tumor. These regions show up with different intensities and areas across different image modalities [2]. The tumor core (TC) subregion shows the bulk of the tumor and is typically removed. The TC contains the enhanced tumor and necrotic fluid-filled (NCR) and the non-enhancing solid parts (NET) of the tumor. These also show up with different intensities across image modalities. The whole tumor subregion

describes the entire tumor since it contains the TC and the peritumoral edema (ED), which is typically depicted by hyper-intense signal in FLAIR.

Given the complexity of the tumor and different image modalities, manual identification of the tumor subregions takes time and is prone to error. Automated methods would facilitate physician diagnosis and lead to better overall patient treatment. A step towards this is the Multimodal Brain Tumor Segmentation (BraTS) challenge [3–5, 5, 2] that invites automated solutions to predict the three tumor subregions from images across four different modalities. It provides 335 patient samples, include 260 HGG cases and 75 LGG cases, each with four MRI modalities: T1, T1 contrast-enhanced (T1ce), T2, and FLAIR. Each image in this dataset has been pre-processed by the same method and rescaled to $1 \times 1 \times 1$ mm isotropic resolution and skull-stripped. The dataset also provides ground truth segmentations of the three subregions. Two additional datasets whose ground truth are unavailable to us are used for validation and test.

Inspired by the success of convolutional neural networks in image recognition tasks, we present a convolutional U-neural network (U-Net) solution to this problem. Our model has multiple encoders for each modality and multiple decoders for each tumor subregion. Below we describe our model in detail, followed by variants of our model and final accuracies on the challenge's validation dataset.

2 Related work

The Convolutional U-Net [6] is the basic architecture for end-to-end semantic segmentation. In previous years of the BraTS competition, researchers have improved upon the basic UNet and addressed training overfitting problems on small datasets. For example, we have 2.5D multi-stage segmentation of different anatomical views [7] and 3D segmentation models [8–10]. The winning entry in the BraTS 2018 contest used an auto-encoder to regularize their network's encoder to prevent over-fitting caused by small sample size [11]. Most teams in the previous contest performed model ensembling or second phase correction to integrate the outputs of multiple models for better final results.

The multi-path approach that has separate encoders for different image types has been explored previously [12]. We have taken this approach further in our previous work [7, 8] with weighted feature fusion blocks to combine features from different modality encoders. That approach works well for for binary segmentation of the brain into healthy and non-healthy tissue. For multiple regions (also known as multi-class segmentation), however, a single fusion block may not work because it uses one set of shared weights for multiple subregions of the brain. The squeeze-and-excite block [13] that we have also used in previous work assigns weights to channel features from different encoders. However, those encoders still fed into the same fusion block. Considering that the different subregions of glioma have different intensities in different modal images, we expect that different subregions will require different weights in the feature fusion stage. Thus we propose a new model with separate fusion and decoder blocks for each subregion of the tumor to be segmented.

3 Methods

3.1 3D convolutional multi-encoder multi-decoder neural network

In Figure 1 we show the overview of our model. We see separate encoders for each of the four image modalities. Each decoder consists of convolutional and transposed convolutional (also called deconvolutional) blocks. We use three decoders corresponding to the three categories: enhanced tumor (ET), tumor core regions (NCR/NET), and the peritumoral edema (ED) within the whole tumor subregion.

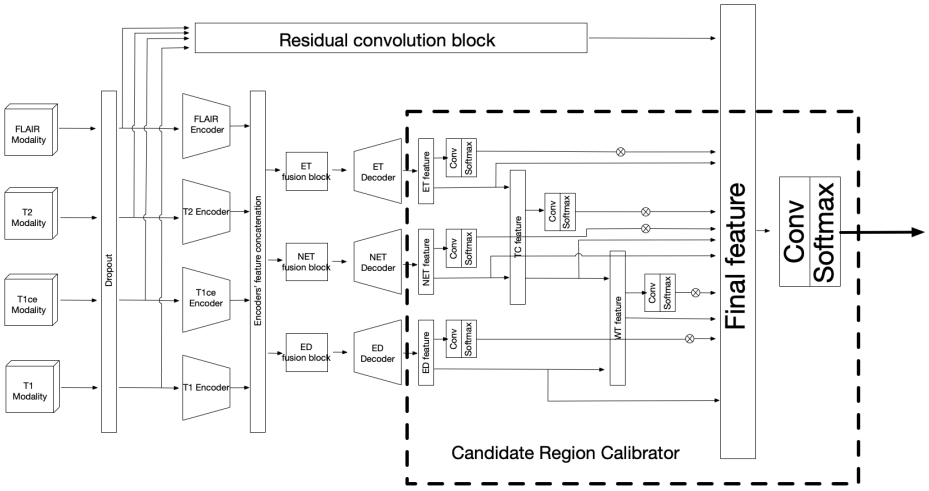


Fig. 1. Our multi-path network with independent encoders for each modality and independent decoders for each tumor subregion

In Figure 1 we see a candidate region calibrator after the outputs of each subregion's decoder. Here we account for the hierarchical relationship between different tumor subregions. We send the output of each decoder to a simple convolutional block (without activation) that outputs a probability. We then multiply the probability by the decoder's output and concatenate it to the outputs of the other three decoders. We concatenate features from the ET and NCR/NET decoders to account for the TC subregion features. We also concatenate TC and ED subregion features to get features for the whole tumor (WT) subregion. We run the outputs from the TC and WT layers through simple convolutional blocks that output probabilities and multiply their outputs by the probabilities.

The purpose of our calibrator is to give specific attention to individual subregions as well as their larger combined parts, which we achieve by multiplying

their outputs by probabilities as described above. The different subregions of gliomas do not necessarily appear in all four modal images. Therefore, we randomly set one of the four modality inputs to zero with probability 0.25 during the training process. We call this modality dropout that we describe in detail below.

Encoder Our overall encoder shown in Figure 2 downsamples the input image four times. In each downsampling we double the number of output filters and send the output to the feature fusion component. The encoder consists of residual convolution blocks shown in Figure 3(c) and downsampling modules in Figure 3(b). We achieve independent encoder training for different modality inputs using group convolution. We use instance normalization as the last component in the residual block so that the output follows a normal distribution. This avoids gradient problems in feature addition part in the decoder (that we identified and solved previously [8]).

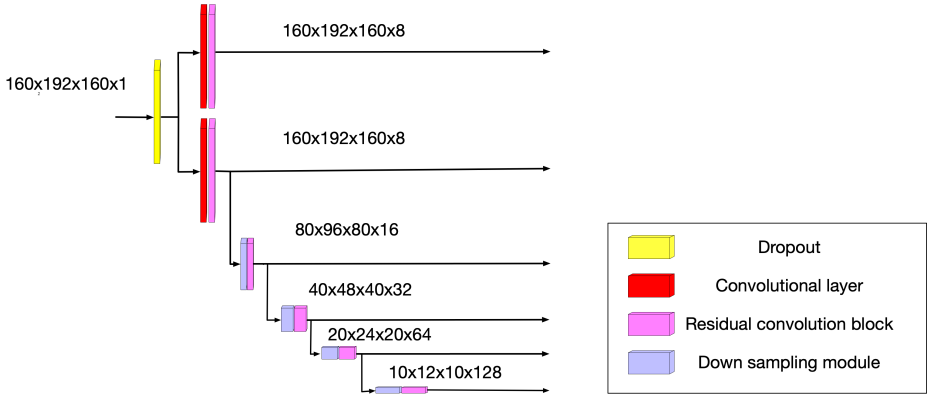
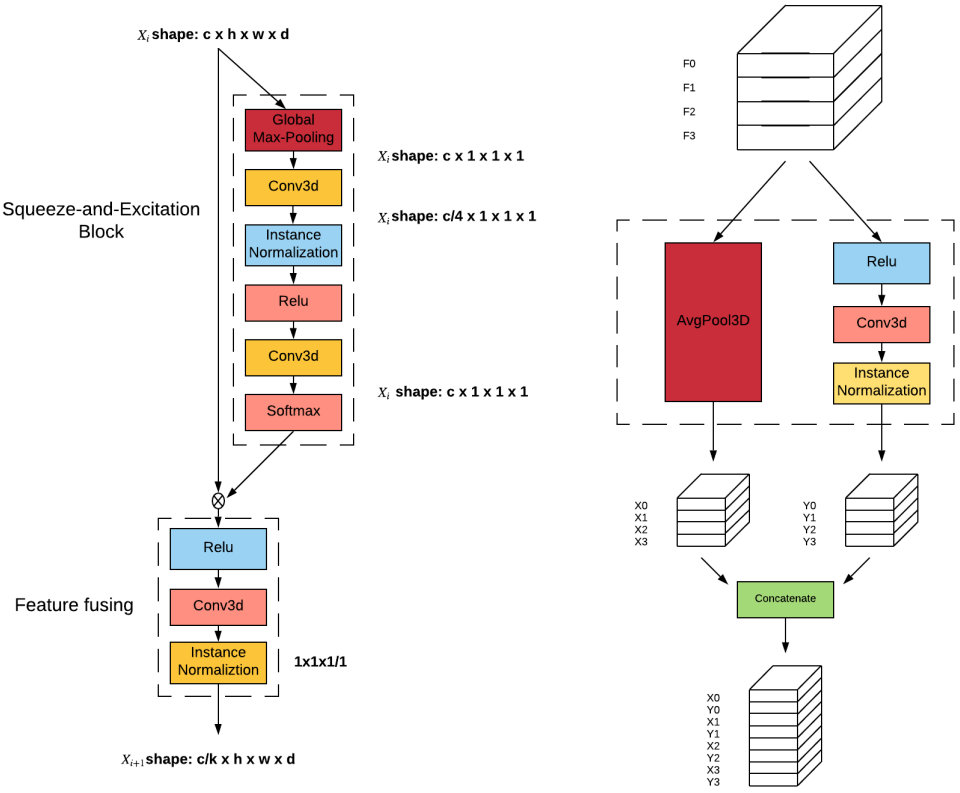


Fig. 2. We downsample the input image four times and collect the output after each residual convolutional block to give to the fusion block.

Feature fusion The output from each downsampling level of the encoder is given to the feature fusion block as shown in Figure 4. The feature fusion block shown in Figure 3(a) integrates features from different modal encoders with a $1 \times 1 \times 1$ convolution. Prior to integration we use the squeeze and effect module to give different weights to the input feature channels.

Decoder The decoder in Figure 5 consists of a residual convolutional block shown in Figure 3(b) followed by a transposed convolutional block (also called upsampling or deconvolutions) shown in Figure 3(d). We add features from the upsampling module to the output of the fusion block before sending them to



(a) Feature fusion block that contains squeeze and excite

(b) Downsampling block

(c) Residual convolutional block

(d) Decoder upsampling block

Fig. 3. We show a detailed description of different components of our network as used in the overall network, encoder, and decoder.

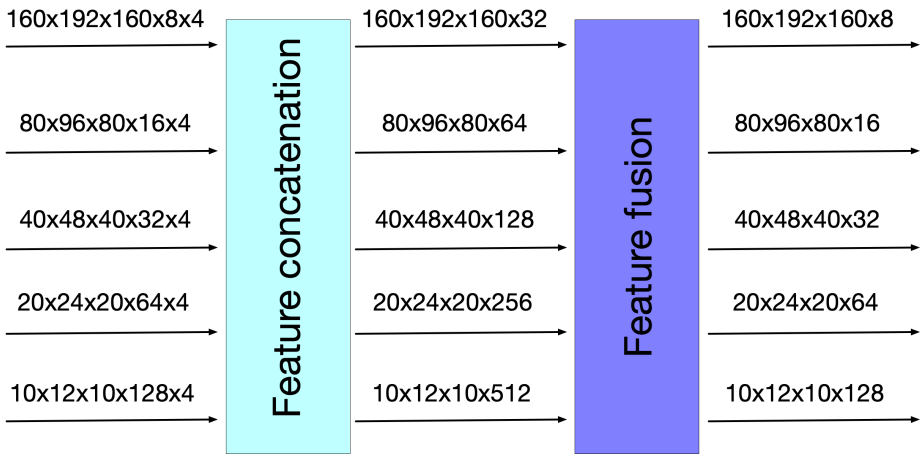


Fig. 4. Features from each downsampling layer of the encoder are collected for fusion. The first set on the left are features from the four encoders for each modality. These are first concatenated and then fused with a 3D convolutional kernel as shown in Figure 3(a).

further upsampling blocks. The output of the decoder is given to the candidate region calibrator.

3.2 Model training and parameters

Loss function We measure the Dice loss of predictions of each the three subregions ET, NCR/NET, and ED after their output from their respective decoder is given to the convolutional block followed by softmax. We also measure the loss of predictions of the TC and WT subregions in the candidate region calibrator after after their outputs are passed through the convolutional block followed by softmax. For a given segmentation the Dice loss is defined to

$$D(p) = \frac{2\sum_i p_i r_i}{\sum_i p_i^2 + \sum_i r_i^2},$$

where p_i are the predicted softmax outputs and r_i is 1 if the voxel has a lesion and 0 otherwise. We also have a final multi-class Dice loss for the three subregions. Our overall loss is the sum of the five Dice losses for the ET, NCT/NET, ED, TC, and WT subregions plus the final multi-class Dice.

Implementation and optimization We implement our network using the Pytorch library [14]. We use stochastic gradient descent (SGD) with Nesterov momentum. We set momentum to 0.9, our initial learning rate to 0.01 and number of epochs at 240. We decrease the learning rate if the current epoch's average loss is no less than the previous epoch's loss. We also use the Pytorch extension

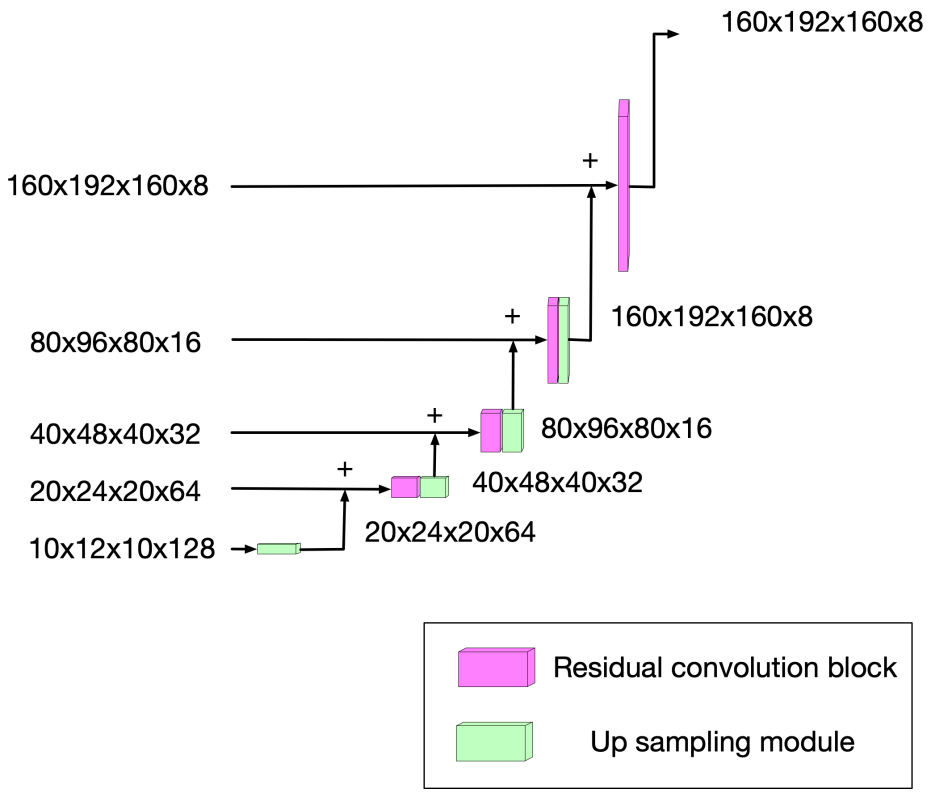


Fig. 5. Features from the fusion block are added to the output of the upsampling layer.

library called NVIDIA-apex for mixed precision (16-bit and 32-bit floats) and distributed training across multiple GPUs [15].

Dropout Dropout is a popular technique to prevent overfitting in neural networks [16]. We perform an image-modality dropout: we randomly pick an image modality and set its input to all zeros. In other words we randomly ignore one out of the four image modalities. One reason for doing this is that some of the tumor subregions are visible only in some modalities. For example enhanced tumor (ET) subregion is visible in T1ce images while the ED subregion in T2 or FLAIR only. We achieve this dropout with the 3D dropout function in Pytorch [14]. Alternatively we could ignore more than one modality but we found this to lower the cross-validation accuracy than when zeroing just one.

Data preprocessing and augmentation We randomly crop each original size image volume from $240 \times 240 \times 155$ to $160 \times 192 \times 144$. We then perform a mean 0 variance 1 normalization (including background zero intensity pixels) for each

modality and each 3D image volume. We randomly flip each 3D image in the three view directions with probability 0.5. In the inference part, we do a center crop of size $160 \times 192 \times 160$ and zero pad to original size without flip.

3.3 Measure of accuracy: Dice coefficient

The Dice coefficient is typically used to measure the accuracy of segmentations in MRI images [17]. The output of our network is a binary mask of the same dimensions as the input image, but with a 1 for each voxel prediced to be a tumor region, and a 0 otherwise. Starting with the human binary mask as ground truth, each predicted voxel is determined to be either a true positive (TP, also one in true mask), false positive (FP, predicted as one but zero in the true mask), or false negative (FN, predicted as zero but one in the true mask). The Dice coefficient is formally defined as

$$DICE = \frac{2TP}{2TP+FP+FN}.$$

4 Results

We first performed a five-fold cross-validation on the training dataset provided by the BraTS consortium. This dataset contains four modality images along with the segmentations of the three tumor subregions. In Table 1 we see the average Dice accuracies and other statistics of our training samples obtained under cross-validation.

Table 1. Average Dice values of our model for each of the three tumor regions after 5-fold cross validation on the training dataset (total of 335 patients).

	Dice			Sensitivity			Specificity			Hausdorff95		
	ET	WT	TC	ET	WT	TC	ET	WT	TC	ET	WT	TC
Mean	0.75	0.90	0.84	0.81	0.90	0.83	1.00	0.99	1.00	5.34	5.67	6.06
Std. dev	0.26	0.07	0.17	0.21	0.09	0.18	0.00	0.01	0.00	10.34	8.31	9.11
Median	0.85	0.92	0.90	0.88	0.93	0.90	1.00	1.00	1.00	1.73	3.38	3.00

We then evaluated our model on the validation dataset provided by BraTS. This dataset contains only the four modality images without ground truth segmentations. To obtain the validation accuracies we uploaded our predicted segmentations to the BraTS server. In Table 2 we see our statistics for the validation dataset returned by the BraTS server. We see that the Dice mean and median values are similar in both training cross-validation and the validation dataset, which shows that our model is generalizing.

To obtain a sense of our method's performance relative to others we evaluate the rank of our validation data Dice accuracies on the BraTS 2019 challenge leaderboard on their web page. At the time of writing of our paper there were

a 113 submissions on all 125 validation samples. We found that on the tumor core (TC) Dice measure our method stood at rank 16 from the top. On the whole tumor (WT) and enhanced tumor (ET) Dice measures our method stood at ranks 22 and 34 respectively. Thus, while not amongst the top three, our method obtained segmentations better than most other submissions.

Table 2. Average Dice values of our model of each of the three tumor regions on the validation dataset provided by BraTS (total of 125 patients)

	Dice			Sensitivity			Specificity			Hausdorff95		
	ET	WT	TC	ET	WT	TC	ET	WT	TC	ET	WT	TC
Mean	0.75	0.90	0.83	0.76	0.93	0.81	1.00	0.99	1.00	5.07	6.13	6.77
Std. dev	0.28	0.08	0.16	0.27	0.07	0.20	0.00	0.01	0.00	12.64	12.18	11.44
Median	0.85	0.92	0.89	0.85	0.95	0.89	1.00	0.99	1.00	2.24	3.16	3.39

In Table 3 we show the average Dice accuracies and the Hausdorff distance [18] on the test data. These were provided to us by the conference organizers since the test data is unavailable to all participants. We see that the average and median Dice accuracies are similar to what we obtained in cross-validation and validation testing above, thus further supporting our model's generalizability.

Table 3. Average Dice values of our model of each of the three tumor regions on the test dataset provided by BraTS

	Dice			Hausdorff95		
	ET	WT	TC	ET	WT	TC
Mean	0.8	0.88	0.83	2.2	4.89	4.09
Std. dev	0.21	0.12	0.24	2.1	5.8	6.8
Median	0.85	0.91	0.92	1.41	3.08	2.45

5 Discussion and Conclusion

We present a multi-encoder and multi-decoder convolutional neural network to handle different image modalities and predict different subregions of the input image (multi-class segmentation). We build upon previous work [8] where we used multiple encoders and squeeze-and-excite blocks [13] to give weights to different modalities. However, in that previous work we used a single feature fusion block that shares weights for different subregions, which is not as accurate as the separate fusion and decoder blocks we developed in this study.

The squeeze-and-excite blocks that we use here are designed for classification. Our global average pooling considers the entire feature map. For tumor subregion segmentation this may not be the best approach since the tumor region is given

by just a small region of the full feature map. Thus in future work we plan to develop squeeze-and-excite to give better modality weights based on just the tumor region instead of the entire feature map.

With 3D components our model is more challenging to train than a 2D one. The additional parameters introduced by 3D typically require more data as well as memory and runtime to train. With the additional parameters the model may overift and so careful training is required. In comparison a 2D model would be easier and faster to train and require less memory but may not be as accurate as a 3D one.

References

1. McKinsey L Goodenberger and Robert B Jenkins. Genetics of adult glioma. *Cancer genetics*, 205(12):613–621, 2012.
2. 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.
3. Bjoern H Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, et al. The multimodal brain tumor image segmentation benchmark (brats). *IEEE transactions on medical imaging*, 34(10):1993–2024, 2014.
4. 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*, 4:170117, 2017.
5. S Bakas, H Akbari, A Sotiras, M Bilello, M Rozycki, J Kirby, J Freymann, K Farahani, and C Davatzikos. Segmentation labels and radiomic features for the pre-operative scans of the tcga-gbm collection. the cancer imaging archive (2017), 2017.
6. Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
7. Yunzhe Xue, Fadi G Farhat, Olga Boukrina, AM Barrett, Jeffrey R Binder, Usman W Roshan, and William W Graves. A multi-path 2.5 dimensional convolutional neural network system for segmenting stroke lesions in brain mri images. *arXiv preprint arXiv:1905.10835*, 2019.
8. Yunzhe Xue, Meiyan Xie, Fadi G Farhat, Olga Boukrina, AM Barrett, Jeffrey R Binder, Usman W Roshan, and William W Graves. A fully 3d multi-path convolutional neural network with feature fusion and feature weighting for automatic lesion identification in brain mri images. *submitted*, 2019.
9. Konstantinos Kamnitsas, Christian Ledig, Virginia FJ Newcombe, Joanna P Simpson, Andrew D Kane, David K Menon, Daniel Rueckert, and Ben Glocker. Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation. *Medical image analysis*, 36:61–78, 2017.
10. Hao Chen, Qi Dou, Lequan Yu, Jing Qin, and Pheng-Ann Heng. Voxresnet: Deep voxelwise residual networks for brain segmentation from 3d mr images. *NeuroImage*, 170:446–455, 2018.

11. Andriy Myronenko. 3d mri brain tumor segmentation using autoencoder regularization. In *International MICCAI Brainlesion Workshop*, pages 311–320. Springer, 2018.
12. Kuan-Lun Tseng, Yen-Liang Lin, Winston Hsu, and Chung-Yang Huang. Joint sequence learning and cross-modality convolution for 3d biomedical segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 3739–3746. IEEE, 2017.
13. Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.
14. Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.
15. Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. Mixed precision training. *arXiv preprint arXiv:1710.03740*, 2017.
16. Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.
17. Alex P Zijdenbos, Benoit M Dawant, Richard A Margolin, and Andrew C Palmer. Morphometric analysis of white matter lesions in mr images: method and validation. *IEEE transactions on medical imaging*, 13(4):716–724, 1994.
18. R Tyrrell Rockafellar and Roger J-B Wets. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.