CSCI 5561 Final Report - Group 12
Tumor Segmentation in Medical Imaging

Bin Hu
hu000562@umn.edu

Tony Zhang
zhan6696@umn.edu

Thomas Reinitz
reini050@umn.edu

1 Introduction & Motivation

Medical imaging plays a critical role in diagnosis, treatment planning, and follow-up of many diseases. One of the most common medical imaging modalities used in clinical practice is magnetic resonance imaging (MRI). MRI provides high-resolution images of soft tissues and organs, making it particularly useful for detecting and characterizing tumors. Tumor segmentation in MRI images is an important task in medical image analysis that can assist radiologists and oncologists in accurate diagnosis and treatment planning. This is important because diagnosing and treating tumors quickly has a significant positive impact on patient prognosis.

The accurate segmentation of tumors in MRI images is a challenging task due to the high variability in tumor shape, size, and appearance. Manual segmentation by radiologists is time-consuming and can be affected by factors such as fatigue and experience. Therefore, the development of automated and accurate tumor segmentation methods is essential to improve the accuracy and efficiency of medical diagnosis and treatment.

In the rest of this report, we cover some previous work on this topic and discuss the methods and compare their advantages and disadvantages. Next we detail our approach to solving the problem. Then we evaluate our experiment results and provide conclusions as to what we learned from the project.

2 Literature Review

Traditional semantic segmentation approaches mostly rely on the hand-crafted features, like texture and colors[5][7]. The benefit of these method is that they do not require a large amount of computation to train, and are relatively intuitive and explainable. In applications where images are very standardized and there is a clear definition of what to look for, these methods can work well. However, the use of hand-crafted features alone is insufficient for accurate tumor segmentation. This approach lacks sensitivity to the context of the image, and a comprehensive understanding of the entire image is necessary for effective tumor analysis. Consequently, traditional methods of semantic segmentation have yielded unsatisfactory results for tumor segmentation tasks.

Fully convolutional network applies the neural network into the task of semantic segmentation (FCN) [8]. It consists of a down-sampling path used to extract and interpret the context and an up-sampling path to localize the correspondence, and achieve better performance on most semantic segmentation tasks. The advantage of this method is that it does not require lots of machinery such as patch-wise training, multi-scale pyramid processing, and ensembles of models. Instead, this method is able to achieve good performance on many kinds of segmentation tasks provided that it has a large quantity of quality training data. The downside to this method is that it takes a lot of high quality data to get good results. This means other methods may out preform FCNs when given less data to work with. Further improvement are implemented on FCN and models like U-Net[11] and SkipDeconv-Net[12] (SD-Net) are developed which shows better performance on medical image segmentation.

Certain following research replace the fully convolutional layers with the Transformer encoder block and the model UNETR[4] achieves state-of-the-art on most medical image segmentation tasks.

3 Approach

3.1 Data Set

We built our tumor segmentation model on the data set of RSNA Brain Tumor Segmentation (BraTS) Challenge 2018[9], which includes clinically-

acquired preoperative multi-modal MRI scans of glioblastoma (GBM/HGG) and lower grade glioma (LGG) in brains. The data set consists of 285 training cases, 66 validation cases, and 191 testing cases. Each case includes approximately 100 depths, 4 MRI modalities (T1, T1c, T2, and FLAIR), and 3 annotations of tumors (whole tumor (WT), tumor core (TC), and enhancing tumor (ET)). Uncompressed, this data takes about 60 GB of space on disk. More recent versions of this challenge data set are available, however the sheer scale of the newer data sets makes computation on available hardware impractical.

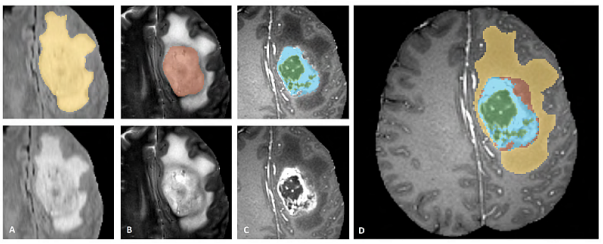

Figure 1: Scan images with the tumor sub-regions highlighted in 3 different modalities (top left) and the final labels(right)[9]

3.2 Model

Two U-Net in different sizes are used in this project: UNET3 and UNET4. UNET3 comprises 3 down-sampling/up-sampling layers and UNET4 comprises 4 down-sampling/up-sampling layers. We expect UNET4 can perform better because it can potentially learn more complex and abstract representations of the input images with its additional layer.

3.3 Optimization Techniques

There are two modes of optimization that we used to improve model performance. First, we would like to improve the generalizability of the model so that it is robust to imperfections in the MRI scan. Some common sources of distortion in MRI images are ghosting or blurring from the patient moving slightly during the scan, and local changes in image brightness due to the patient having medical devices in the room [3]. To address this, we will randomly introduce blurring, noise, slight translation and/or slight rotation to the training data, adding these changed examples to the original training data before testing, known as **Data Augmentation**. By performing this augmentation, we avoid having to use other regularization techniques that may adversely impact the performance of our model.

Care was taken to ensure that the augmentation did not leave behind artifacts such as blank spaces after rotation and translation. Additionally, to be more accurate to real world conditions, only 25% of the data is augmented. This should force the model to learn mostly on good data but, be able to handle some abnormalities as well.

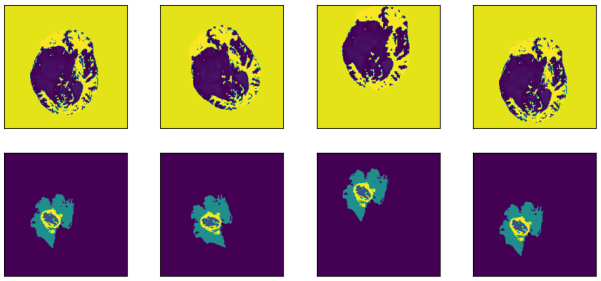

Figure 2: Original image channel on the left with three different augmentations on the right. The parameters were turned way up for demonstration purposes so that it is clearly visible what changes have occurred and that there are no visible artifacts.

Secondly, to improve the classification ability of the model, we created a denoising pipeline for the inputs, so as to train the network on clearer and cleaner inputs. There are a few ways to do this, and we tested two main methods, as described in the next paragraph. The main key for denosing in this context is to clarify and keep sharp edges for segmentation, while improving image clarity. There were two metrics used to compare the denoising methods: PSNR (peak signal-to-noise ratio) and SSIM (structural image similarity). The SSIM is much more telling than the PSNR, as it provides a good way to analyze how well the structure of the image is kept. PSNR, however, is a more outdated method of measure that scores in a similar way that MSE would score on an image, ignoring edge clarity, illumination, and other parts and structures of an image. The synthetic noise used was a combination of gaussian noise and multiplicative gaussian noise (i.e. "speckle" noise), two common medical noises, the latter of which being a particularly difficult type of noise to get rid of by traditional methods.

Anisotropic diffusion [10] is an older method for denoising while keeping edge clarity, it computes and uses gradients for diffusion, reducing diffusion near sharper gradients (edges). This implementation was based on the python translation of the research paper by Alistar Muldal [1], with some im-

plementation differences and an extra sharpening step using the laplacian to try and preserve edges. It performes well enough on the dataset, with a PSNR score of 39.07, and an SSIM score of 87.69% similarity. However, the SSIM score shows that the image was losing significant structural clarity, and would likely perform worse in segmentation tasks. This was likely due to the diffusion oversmoothing near soft edges, rather than perserving them. Also, a significant trouble was the multiplicative noise, it is difficult for AD to smooth over the "hard edges" created by multiplicative noise without oversmoothing the edges.

An improvement made to the denoising pipeline was the method of CNN denoising, with architecture similar to CBDNet [2]. The CBDNet implmementation for our denoising method uses a 4 layer noise estimation subnetwork, connected to a larger non-blind denoising network. L2 loss was used, and Adam with learning rate decay of 0.8 and 2 trained epochs. Instead of ReLU as the activation function, an approximation to the GeLU activation was used as the activation function for the hidden layers. It was trained using the idea of Noise2Noise [6]; the latent space of the image is the closest thing between two noisy images, so by training on a noisy image to the same image with different noise (added synthetic noise), we can estimate the denoised image. There were a low number of epochs due to a combination of factors, including lack of computing power, and preventing the network from learning the noise tranformation rather than the latent image space. The final test set gave a PSNR of 42.45 and an SSIM of 93.80% similarity, a significant uplift in performance from both metrics in comparison to the AD method. Importantly, the structural similarity was much improved from the AD method, meaning significant edge clarity was kept while denoising.

Denoising Method	**PSNR**	**SSIM**
Anisotropic diffusion	39.07dB	87.69%
CBDNet style denoiser	**42.45dB**	**93.80%**

Table 1: Results of denoising method on test set (best shown in bold). As shown, NN performs better in both metrics, but significantly so in SSIM

3.4 Evaluation

The evaluation metric used to compare models was Dice score.

Dice score Dice score is a measure of similarity between two sets of data, usually used in medical image analysis. It ranges from 0 to 1, with 1 indicating a perfect match between the two sets of data.

$$\text{Dice Score} = \frac{2|A \cap B|}{|A| + |B|}$$

Where A and B are two sets of points being compared.

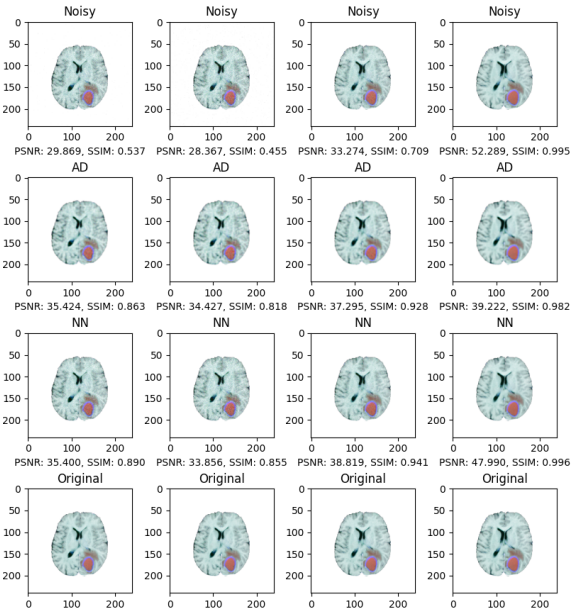

Figure 3: A slice of the denoising results. AD is anisotropic diffusion, and NN is the CBDNet-style neural network. As shown in the slice, the Neural Network provided a boost to image quality comparative to AD

4 Experiment and Results

We trained 4 models in this project, they are:

(1) UNet3 with transformation
(2) UNet3 without transformation
(3) UNet4 with transformation
(4) UNet4 without transformation

The models are trained in the same configuration which is listed in Table 2. A checkpoint of the model will be saved after each epoch and the checkpoint with the lowest loss on valid set will be employed to calculate the Dice Score.

The changes in validation loss for the four models during the training process are illustrated in Figure 3, and the Dice Score for all models are illustrated in Figure 4.

The highest overall score is achieved by the UNet3 model with transformation. And the per-

Epochs	6
Batch size	16
Optimizer	AdamW
Scheduler	Cosine scheduler
Max Learning Rate	1e-5
Min Learning Rate	1e-6
Warm-up Step Ratio	0.1

Table 2: Training configuration

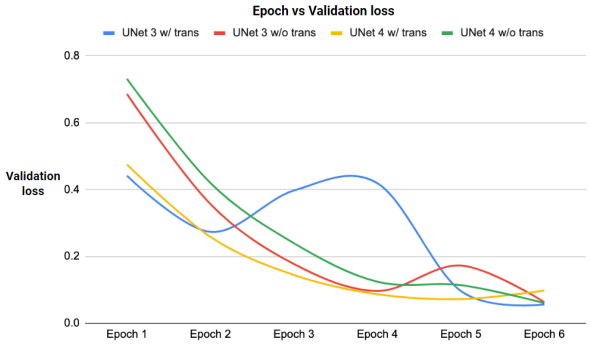

Figure 4: Epoch vs Validation loss during training for all four models trained.

Dice Score of	UNet 3 w/ trans	UNet 3 w/o trans	UNet 4 w/ trans	UNet 4 w/o trans
Necrotic and non-enhancing tumor core	**0.526**	0.483	0.507	0.474
Peritumoral edema	**0.584**	0.552	0.474	0.541
GD enhancing tumor	0.605	**0.654**	0.498	0.54
Total	**0.573**	0.553	0.487	0.524

Figure 5: Dice score for all models and types of tumor detected. The best scores are highlighted in blue.

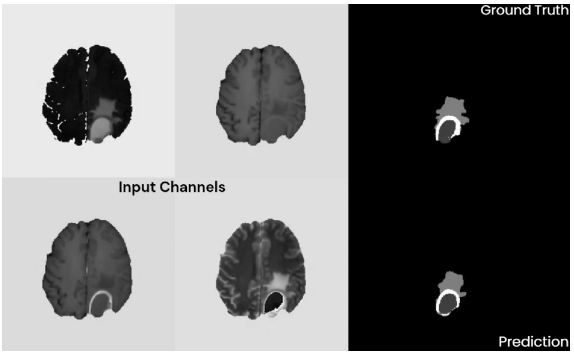

Figure 6: One frame of the visualized prediction

formance of UNet3 generally exceeds that of the UNet4 model.

Our augmentation strategy (transformation) has helped the UNet3 model to generalize better on the validation set and achieve a higher score. However, the results show that our initial assumption that a deeper network would always perform better is incorrect. One explanation is that our UNet4 model is easier to overfit due to its increased complexity compared to the UNet3 model.

Moreover, we observe that the Dice Score of UNet4 model with transformation augmentation is worse than UNet4 model without transformation. We believe that this is due to the underfitting of the UNet4 model with transformation, as the transformation causes the model to converge more slowly while being less susceptible to overfitting. After 6 epochs, the UNet4 model has already shown signs of overfitting, whereas the UNet4 model with transformation has not yet converged. We speculate that training for additional epochs may produce more compelling results.

We have visualized one example prediction made by the best model (UNet3 with transformation) on one valid case as video. Here is the link of the video:

https://www.youtube.com/watch?v=U-xrFIYS0ho

5 Conclusions

Medical image segmentation is a valuable application in computer vision because automatic diagnostic tools can greatly assist medical practitioners in the speed of treatment. In this project, we created four computer vision models to detect brain tumors in MRI scans. We used data from the BRATS 2018 Competition [9]. The models were two UNet3 models, and two UNet4 models with one of each being assisted by optimization techniques to try and enhance model performance.

After training and testing our models, the UNet3 with optimization methods had the best overall score achieving dice score of 0.573 on the validation set. In general, the UNet3 performed better than UNet4, and the optimization seemed to make the UNet4 perform worse. Visual inspection of the model's predictions reveal that our best model is able to mirror the ground truth labeling accurately, however it struggles with identifying sharp boundaries and is a little smoother than the real label. Finally, our UNet4 model with transformations shows that it has not converged despite UNet4 without transformation showing signs of convergence. This lead us to theorize that the UNet4 with transformation may have under fit to the data and could have learned more complex patterns in the

segmentation task with more epochs.

One limitation of our work is that we could not run our models against the test data of the competition so we cannot assess our performance on the leaderboard. Another limitation is that due to the training time of the models we were only able to run six epochs per model. Since is possible that we underfit to the data, potential future world could be running more epochs on the models to let them realize their full potential.

6 Contributions of Team Members

Bin completed the framework of the code and implemented the two different size of U-Net. He trained the four models and collected the data. Bin learned that a deeper network does not necessarily improve the performance of the model by comparing the U-Net in different sizes.

Tony completed the data augmentation section, as well as creating visualizations for the report and the presentation. Tony learned that data augmentation can easily make the model preform worse if not done right. In the end, it is best to find a balance between model generalizability and performance.

Tom completed the denoising method and some of the optimization section, as well as creating visualizations for presenting the denoising methods. Tom created a denoising pipeline, testing two different types of denoiser, with 3 different models (the initial try for NN denoising was a simple autoencoder model; It was not talked about extensively in this paper, as it was mainly a proof of concept model before using more complex CNN models). Tom learned that knowing the likelihood, typical amount, and especially typical type of noise on the dataset makes a big difference in the methods to use in denoising and their performance. Speckle noise is much more of an issue to denoise than typical gaussian noise, and the typical methods to reduce gaussian noise don't work super well with speckle noise. Also, edge clarity in the denoising methods are much more important to the image segmentation than overall L2 loss pixel difference

References

[1] Alistair Muldal. Anisotropic diffusion python implementation.

[2] Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. 2019. Toward convolutional blind denoising of real photographs. *2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[3] Vikki Harmonay. 2020. Most common artifacts in mri.

[4] Ali Hatamizadeh, Yucheng Tang, Vishwesh Nath, Dong Yang, Andriy Myronenko, Bennett Landman, Holger R Roth, and Daguang Xu. 2022. Unetr: Transformers for 3d medical image segmentation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 574–584.

[5] Dana E. Ilea and Paul F. Whelan. 2011. Image segmentation based on the integration of colour–texture descriptors—a review. *Pattern Recognition*, 44(10):2479–2501. Semi-Supervised Learning for Visual Content Analysis and Understanding.

[6] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. 2018. Noise2noise: Learning image restoration without clean data.

[7] Geng-Cheng Lin, Wen-June Wang, Chung-Chia Kang, and Chuin-Mu Wang. 2012. Multispectral mr images segmentation based on fuzzy knowledge and modified seeded region growing. *Magnetic Resonance Imaging*, 30(2):230–246.

[8] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2014. Fully convolutional networks for semantic segmentation. *CoRR*, abs/1411.4038.

[9] Bjoern H. Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, Levente Lanczi, Elizabeth Gerstner, Marc-André Weber, Tal Arbel, Brian B. Avants, Nicholas Ayache, Patricia Buendia, D. Louis Collins, Nicolas Cordier, Jason J. Corso, Antonio Criminisi, Tilak Das, Hervé Delingette, Çağatay Demiralp, Christopher R. Durst, Michel Dojat, Senan Doyle, Joana Festa, Florence Forbes, Ezequiel Geremia, Ben Glocker, Polina Golland, Xiaotao Guo, Andac Hamamci, Khan M. Iftekharuddin, Raj Jena, Nigel M. John, Ender Konukoglu, Danial Lashkari, José António Mariz, Raphael Meier, Sérgio Pereira, Doina Precup, Stephen J. Price, Tammy Riklin Raviv, Syed M. S. Reza, Michael Ryan, Duygu Sarikaya, Lawrence Schwartz, Hoo-Chang Shin, Jamie Shotton, Carlos A. Silva, Nuno Sousa, Nagesh K. Subbanna, Gabor Szekely, Thomas J. Taylor, Owen M. Thomas, Nicholas J. Tustison, Gozde Unal, Flor Vasseur, Max Wintermark, Dong Hye Ye, Liang Zhao, Binsheng Zhao, Darko Zikic, Marcel Prastawa, Mauricio Reyes, and Koen Van Leemput. 2015. The multimodal brain tumor image segmentation benchmark (brats). *IEEE Transactions on Medical Imaging*, 34(10):1993–2024.

[10] P. Perona and J. Malik. 1990. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639.

[11] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for

biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer.

[12] Abhijit Guha Roy, Sailesh Conjeti, Debdoot Sheet, Amin Katouzian, Nassir Navab, and Christian Wachinger. 2017. Error corrective boosting for learning fully convolutional networks with limited data. In *Medical Image Computing and Computer Assisted Intervention MICCAI 2017*, pages 231–239, Cham. Springer International Publishing.

A Code repository

The code is available at our GitHub repository:
https://github.com/binhu02/CSCI5561_Project

B Video of prediction visualization

The video for the prediction visualization is available at Youtube:
https://www.youtube.com/watch?v=U-xrFIYS0ho