

GU-Net: Diffuse Glioma Segmentation in Brain MRIs Using a Modified U-Net Under Data Constraints

Arnav Dhar¹, Snehil Kakani[#] and Caroline Hsu[#]

¹Northwood High School, USA

[#]Advisor

ABSTRACT

Diffuse gliomas are a prevalent type of brain tumor in adults. Currently, treating these tumors is a time-consuming process. Radiologists manually identify and segment diffuse gliomas in Magnetic Resonance Images (MRIs), which are then used as reference by surgeons during treatment. Prior research conducted on automating this process utilizes machine learning (ML) models such as CNNs and U-Nets. One key piece of prior work is BU-Net, which slightly alters the architecture of U-Net. To contribute to this field, we propose a novel, simplified version of BU-Net, dubbed GU-Net, optimized specifically for low-computation neuroimaging. The proposed model is trained on a subset of the BraTS 2021 dataset, consisting of a mere 1647 images stemming from 549 different brain MRIs. Under data constraints, we achieve a 71.58% dice similarity coefficient (DSC) and 64.29% Intersection Over Union (IOU) on the testing dataset. Compared with U-Net's 0.672 and 0.611 and BU-Net's 0.613 and 0.554 on the same dataset, GU-Net's success under data constraints compared to the other two models is shown. Our work specifically advances diagnosis in underprivileged areas and hospitals with less funding, as GU-Net requires less data to be used and has higher efficiency compared to existing solutions. Dataset: <https://www.kaggle.com/datasets/dschettler8845/brats-2021-task1>

Introduction

Diffuse gliomas account for nearly 80% of all malignant brain tumors in adults and often require precise surgical treatment (Finch et al). The segmentation of brain tumors from MRI images is essential, as accurate segmentation helps facilitate accurate classification. The segmentation of these MRI images involves processing the MRI images with the tumor, outlining the tumor, then returning an image of the tumor without the surrounding material. Conventionally, radiologists manually identify and annotate MRI images, a process that takes manual labor and time. This is especially true in the case of diffuse gliomas due to their unclear borders (as seen in Figure 1) and low contrast in raw MRI images (Havaei et al). Hence, the issue of human error becomes prominent. To solve this issue, the use of machine learning has risen as a solution to optimize this task (Fletcher-Heath et al). Early attempts at automizing tumor segmentation used primitive thresholding and edge detection techniques. However, such methods, such as Sobel or Canny edge detectors, often achieve low DSC and IOU scores due to their struggle with irregularly shaped tumors.

Current methods for brain tumor segmentation often use convolutional neural networks, or CNNs, because of their ability to capture spatial hierarchies in visual features (Agrawal et al). One popular CNN that has been used to segment biomedical images is U-Net. U-Net uses an encoder-decoder framework, resulting in an output image with the same dimensions as the input, as seen in Figure 2. The model encodes the inputs by applying same-convolution and max pooling layers to downsample the image. Then, the inputs are upsampled via transposed convolution layers. These layers take a kernel and image matrix as input to expand the image

matrix using a convolution matrix, producing a segmentation mask. There are also skip connections in between blocks across from one another, allowing the mitigation of information dilution and stabilization of gradients. This network was revolutionary for segmentation tasks because it had a much higher DSC and IOU compared to previous networks at the time.

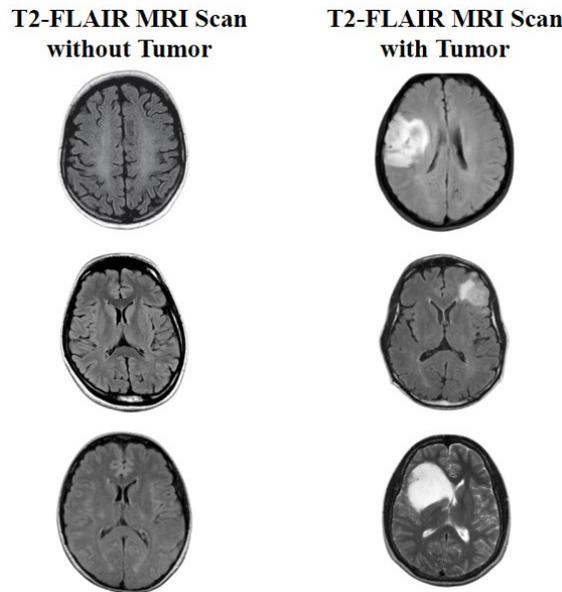


Figure 1. Differences between MRI scans with and without tumors.

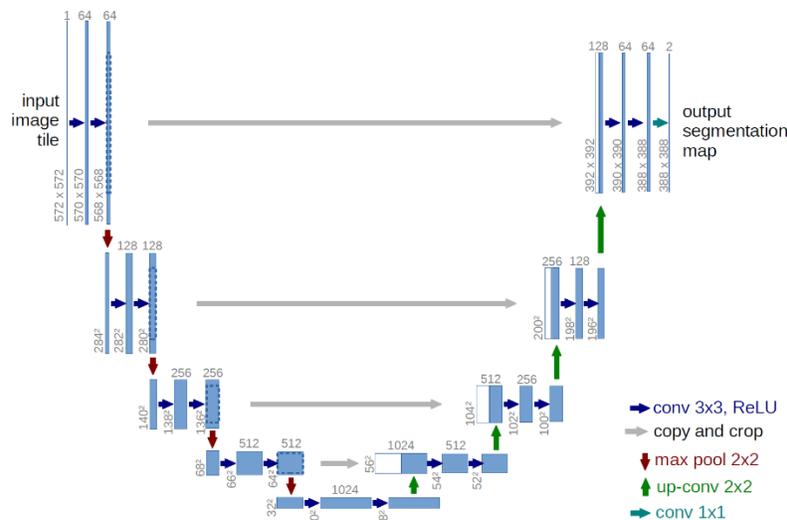


Figure 2. The Original U-Net Architecture

U-Net was originally developed to solve a cell tracking problem; however, the architecture was quickly adapted to solve other tasks (Ronneberger et. al). The first instance of the implementation of U-Net for brain segmentation was done in 2017 by Dong et al. Their model was trained from the BraTS 2015 dataset, a dataset consisting of 220 brain MRIs with tumors. After they modified the U-Net architecture to suit brain tumor segmentation, their model achieved a dice similarity coefficient (DSC) of 0.81 (Dong et al). Dong et al. utilized a

total of 27 layers and used a maximum of 100 epochs in order to fully train the model. Another significant application of U-Net was done in 2020 by Wang et al. Wang et al. utilized a 3D U-Net architecture to segment tumors using two different patching strategies: dividing the whole augmented image into equally sized patches for the first strategy, and dividing it into successively smaller patches for the second one. Wang et al. trained off of the BraTS 2019 dataset and used 100 epochs to achieve 0.894 and 0.852 DSC for segmenting the whole tumor in the validation and testing sets respectively. In 2020, Rehman et al developed a neural network called BU-Net that performs a multiclass semantic segmentation with gliomas, dividing it into the whole tumor, tumor core, and enhancing core. (Rehman et al) Their model trained from the BraTS 2017 and BraTS 2018, improving off of previous state-of-the-art methods and achieving a DSC of 0.901 for the whole tumor.

A common theme present in these deep neural networks is the high volume of computational power and data used to train the model. For instance, the BU-Net architecture uses around 65,000 images to train the model. The BraTS 2017 and BraTS 2018 dataset combined requires around 8 GB of space (Baid et al). Processing through all the images and training the model requires high amounts of computational power because of the 30 million parameters the model has to learn (Ali et al). This becomes an issue for low-funded hospitals because they lack the resources, such as GPU and storage space needed to utilize models effectively. Large amounts of training data are also not always available, especially in the context of healthcare. Therefore, GU-Net was developed to better understand the performance of the U-Net and BU-Net architectures when supplied with low amounts of data for training, validation, and testing. GU-Net uses only 22 million trainable parameters and utilizes only 62 epochs to train our model. It removes two layers from the BU-Net architecture, uses different pooling techniques, has more filters, and implements L2 regularization. Even with these constraints, GU-Net demonstrates the learning capabilities of these models and the potential for high-performing networks that do not require large datasets. After training the GU-Net model, our model achieved a test DSC of 0.716 which is comparable to the U-Net architecture used by Dong et al and Wang et al, considering the difference in training images GU-Net used compared to U-Net and BU-Net (see Table 1 for quantitative comparison). The training processes for U-Net and BU-Net are computationally expensive, especially in the field of medical imaging. Therefore, when large models such as U-Net and BU-Net are modified and trained via limited data and epochs, the computational burden is reduced and the training process is made more efficient.

Methods

Dataset

We used a subset of the BraTS 2021 training dataset. Although the dataset provided 4 different MRI modalities – T1, T1-Post Contrast, T2, and T2-Flair – it was found that T2-Flair to be optimal due to its high contrast and clear visibility of the tumor, as shown in Figure 3. After filtering out all other imaging modalities from the dataset, a total of 1251 3D brain MRI scans in the NIfTI file format was reached, each with 3 different view-points, as shown in Figure 4. Since our model uses 2D convolutional layers, only choose images from the axial view were used since the axial view is the primary angle used by doctors when identifying tumors in MRI scans. The model only used a subset of provided brain MRIs, approximately 549 scans to demonstrate that GU-Net can be optimized for limited data. Then, to augment the data, the center 3 slices per brain MRI were taken, giving a final total of 1647 images. It is important to note that the MRI scans were split into training, validation, and testing in a 80/10/10 ratio before the scans were spliced to ensure that slices from the same MRI will not be present in training and validation/testing.

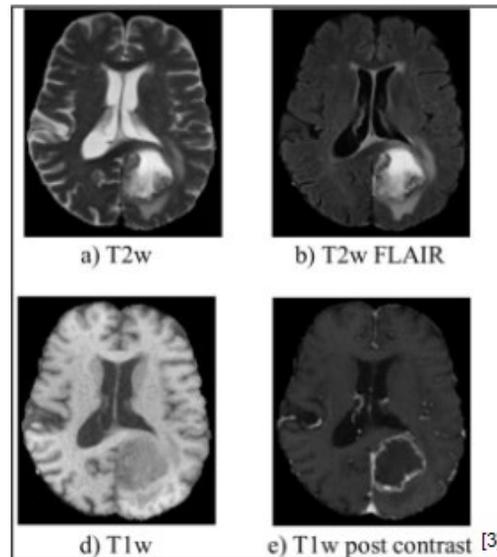


Figure 3. The different MRI modalities, visualized

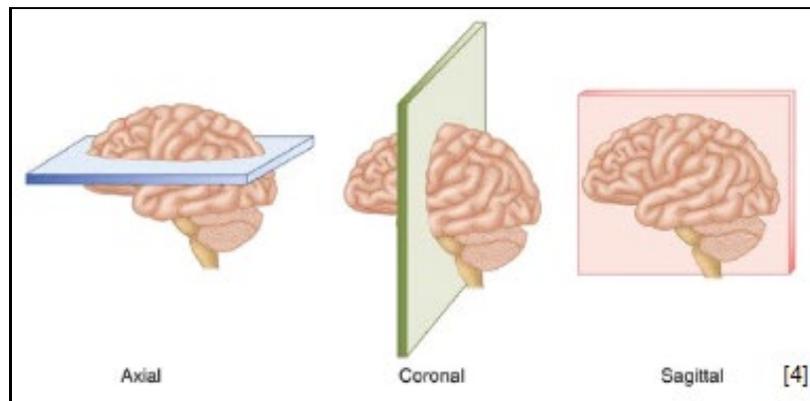


Figure 4. The top, frontal, and side view of the brain, provided in each MRI scan.

Image Preprocessing

Image pre-processing is necessary in order to prepare the data for efficient usage. Because U-Net is trained on two-dimensional data, we first converted our data from three-dimensional NIfTI files to two-dimensional image slices by taking the center 3 slices of each image. This resulted in a total of 1647 images. We normalized the intensity values of each image by clipping the range to be 0 to 1. Then, we preprocessed each image's pixel intensity values to clean the data. We started by normalizing the intensity values of each image in order to increase the efficiency of our model. Because intensity values in some of the files were negative, we subtracted the minimum pixel intensity value of each file from every pixel in that file. This caused every intensity value to be a positive number; this then allowed us to divide each intensity value in each image by the maximum intensity value in each image, hence clipping the intensity values per image to be from 0 to 1. Then, we set a lower threshold value for pixel intensity in order to filter out any regions that had an intensity value but were not bright enough to be part of a tumor, lowering the size of each file. We used a threshold value of 0.3, which we observed to be an effective threshold for maintaining the whole tumor while removing unnecessary parts of

the image. In order to generalize the data, we rotated the images 0, 90, 180, and 270 degrees in a 1/1/1/1 ratio. The generalization of data helps avoid overfitting.

Architecture

Our model architecture is derived from Rehman et al’s BU-Net, which expanded upon the original U-Net architecture with two additional building blocks: the WC and RES block. The BU-Net model structure is depicted in Figure 5. The WC block consists of two convolution layers with one dimensional kernels. Its primary usage is to expand the output into a larger channel size, as the data passes through it just before upsampling begins. Meanwhile, the RES block is similar to the WC block, but includes various convolution layers that decrease in kernel size, as well as a skip connection that is concatenated to the added output of all convolution layers. The RES block is inserted in the copy bridge between the downsampling and upsampling layers. The novelty in GU-Net compared to BU-Net, as depicted in Figure 6, is primarily the reduction of layers. To combat overfitting on the limited data, we removed two encoder-decoder layers from the architecture. Additionally, we implemented an L2 weight regularization, or weight decay, of 0.01. Thus, we felt that there was not a need for dropout after each double convolution due to these modifications. However, we did increase the number of kernel channels, or filters, to better understand the data compared to the BU-Net architecture. To better suit the type of data we had, we decided to use average pooling over max pooling simply because average pooling retains more data compared to max pooling, which we found was crucial for our model to train better. We also used the Xavier normal weight initialization algorithm to initialize all of the convolution and transposed convolution layers in GU-Net whilst BU-Net uses He initialization. Our implementation of the model architecture was designed in the Pytorch programming framework.

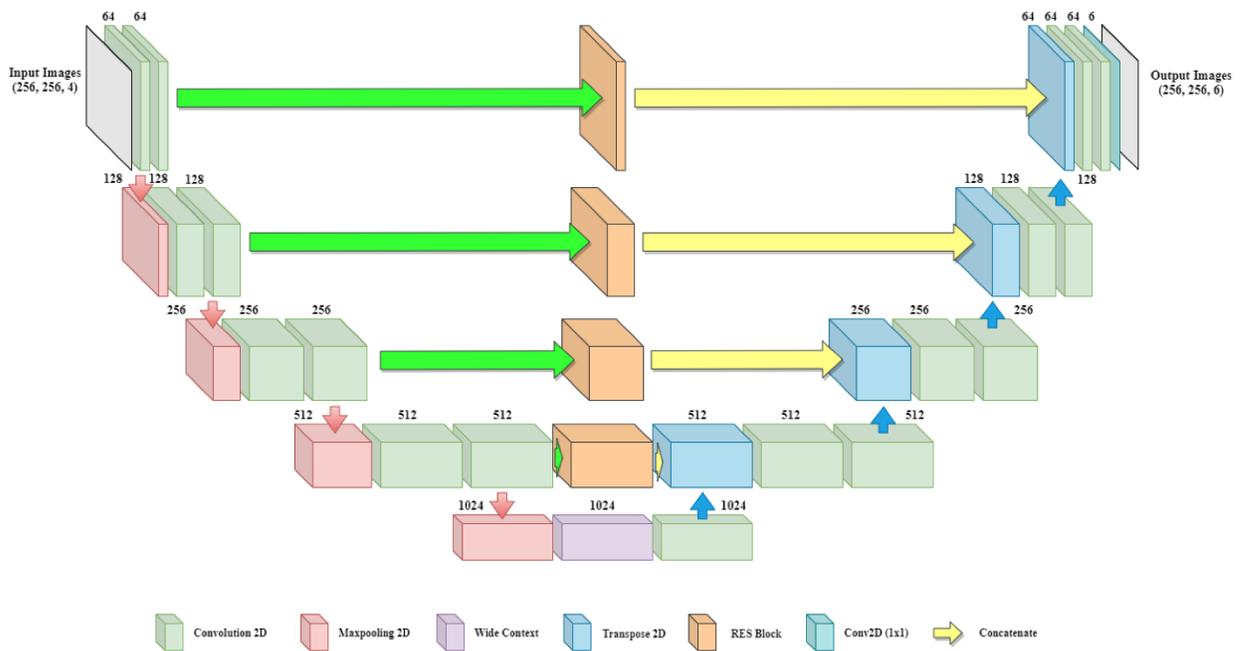


Figure 5. BU-Net Architecture

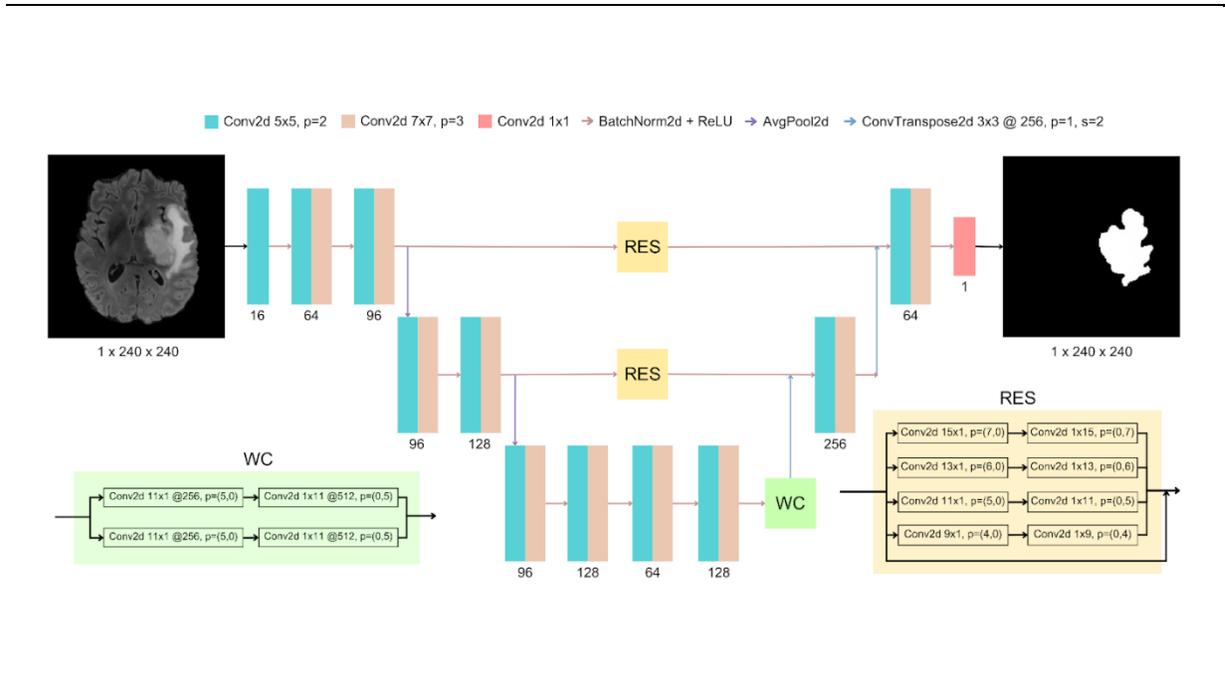


Figure 6. Proposed GU-Net Architecture

Training

Training GU-Net was fairly straightforward. We used a simple training loop to gradually update weights over 62 epochs. We found that using the Adam optimizer in conjunction with a learning rate of 0.001 yielded the best overall results, as the algorithm decreases the learning rate over time as the model learns more information to prevent overfitting. As previously mentioned, a weight decay of 0.01 was also used to reduce overfitting. We developed a custom loss function composed of the dice similarity coefficient converted into a loss function by subtracting it from 1 and binary cross entropy with Logits, added together. The dice loss was weighted at 0.9, while the BCE loss was weighted at 0.1 because we decided that due to the segmentation problem we were solving, dice would be a more important metric to define loss by, and BCE was not as important. The loss function is depicted in (1). Our training loop included the training batch iteration, which took in data in shuffled batches of 8 and ran a forward and backward pass, and the validation batch iteration, which used validation data in shuffled batches of 1 to evaluate the performance of the model. We then saved the model which had the highest validation dice similarity coefficient at that point, creating an “early stop” in the training once the validation got to the highest point. Finally, after every 10 epochs, the model was tested on unseen data in order to evaluate the final dice similarity coefficient and intersection-over-union.

$$L(X, Y) = 0.9 \left(1 - \frac{2|X \cap Y|}{|X| + |Y|} \right) + 0.1(-Y \log(X) + (1 - Y) \log(1 - X))$$

Equation 1: Mathematical representation of weighted Dice-BCE loss function.

Results

$$Dice = \frac{2|X \cap Y|}{|X| + |Y|}$$

Equation 2: Mathematical representation of the Dice Similarity Coefficient.

In order to evaluate the effectiveness of our model, we used the dice similarity coefficient (DSC). The DSC is used by previous state-of-the-art models, making it the best method to compare the results of our architecture to other neural networks. The DSC is represented in Equation 2. The DSC gives weight to the intersection between the two separate areas (represented as X and Y) and divides by the total area between the two areas. In this case, we compared our predicted segmentation mask with the given segmentation mask from the dataset.

Table 1. Evaluation of GU-Net model to evaluation of BU-Net and U-Net with no data constraints

	Dataset Used	# of Training Slices	Test Data Dice Similarity Coefficient	Test Data Intersection Over Union	# of Parameters
Our Model (GU-Net)	BraTS 2021	45,210	0.816	0.723	~22.6 million
U-Net	BraTS 2021	45,210	0.860	0.816	~30 million
BU-Net	BraTS 2021	45,210	0.901	0.874	~81.3 million

Table 2. Evaluation of GU-Net model to evaluation of BU-Net and U-Net with data constraints

	Dataset Used	# of Training Slices	Test Data Dice Similarity Coefficient	Test Data Intersection Over Union	# of Parameters
Our Model (GU-Net)	BraTS 2021	1,320	0.716	0.643	~22.6 million
U-Net	BraTS 2021	1,320	0.672	0.611	~30 million
BU-Net	BraTS 2021	1,320	0.613	0.554	~81.3 million

Table 3. Table displaying results of evaluation of GU-Net’s Train, Test, and Validation split

	# of Slices	Dice Similarity Coefficient	Intersection Over Union
Training	1,320	0.927	0.864
Testing	165	0.716	0.643
Validation	162	0.752	0.627

Table 1 shows GU-Net's testing DSC compared to BU-Net and U-Net when given the same amount of training data as those models. The BU-Net and U-Net model code was obtained, and the models were trained and evaluated on the BraTS 2021 dataset along with GU-Net. Adding on, Table 2 shows the models' performances when operating with merely a subsection of the original dataset. The fact that the DSC and IOU of GU-Net dropped only ~10% under data constraints compared to the ~20% of U-Net and ~30% of BU-Net demonstrates GU-Net's proficiency at operating under data constraints. Taken together, the two tables show the relationship between the data constrained GU-Net and the normal U-Net and BU-Net. With only 2.9% of the training data that the normal U-Net used, and 2.0% of the training data that normal BU-Net used, the data constrained GU-Net achieves 88.4% of U-Net's DSC and 79.7% of BU-Net's DSC. This demonstrates GU-Net's proficiency in achieving meaningful results even when trained on very low amounts of data, prompting further work testing GU-Net's capabilities when given amounts of training data comparable to that of current state-of-the-art models.

GU-Net was developed and trained solely on one GPU to continue with its goal of using minimal resources so that it could be used in underprivileged areas. However, if given the time and computational resources that current state-of-the-art models use, a boost in performance would likely have been observed.

Table 3 represents the DSC and IOU GU-Net achieved for training, testing, and validation. It also displays the 80/10/10 data split used to train, test, and validate the model. Because the training DSC and IOU of the training data are significantly higher than that of the testing and validation data, we can assume that the model overfit on the training data and lost some of its ability to generalize and account for unseen data. This phenomenon can usually be combated by further regularization to further simplify the model, meaning that increasing L2 strength or adding a dropout layer may be necessary for further hyperparameter tuning.

Conclusion

In this paper, we proposed, created, and tested a novel neural architecture, dubbed GU-Net, that is specifically optimized for brain tumor segmentation. Our model is inspired from the BU-Net architecture; however, we changed the architecture to fit our data constraints by deleting 2 layers and simplifying the convolutional layers. Using a subset of only 1647 images from the BraTS 2021 training dataset, we split the images into a 80/10/10 ratio for training, validation, and testing. Through this, GU-Net proves to have comparable results to existing state-of-the-art methods.

One limitation GU-Net faces is the lack of training with brain MRI images without tumors. The model was trained solely on images with tumors, so when a brain MRI without a tumor is introduced, GU-Net will produce a mask of a nonexistent tumor. Another limitation GU-Net faces is worse DSC and IOU with smaller tumors. This phenomenon is due to its simplified architecture. Because of the smaller number of trainable parameters in GU-Net, it produces segmentations with less detail than conventional state-of-the-art models; therefore, it sometimes enlarges or skips over small tumors.

The structure of the model is proficient, but can certainly be improved upon in the future. One immediate plan to improve DSC and IOU is to continue changing the configuration of GU-Net, specifically by adding dropout layers after each pooling layer. Evaluating the model on other Kaggle datasets consisting of brain MRI scans to ensure its ability to generalize would be vital. Additionally, in the future we would test the model with other MRI formats that came with this dataset that were not used to train the model, namely T1, T2, and T1+C. Testing done with different MRI formats can improve the model's DSC, and eventually training the model to be able to segment via multiple different formats would improve its usability and generalization.

Author Contribution Statement

A.D. contributed to writing and editing the Abstract, Introduction and Dataset sections and fully wrote the Image Preprocessing and Results sections. A.D. also contributed to literature review, created most of the figures in the manuscript and presentation, and wrote the loss function and DSC functions in the code.

C.H. contributed to writing the Abstract, Conclusion, and majority of the Introduction and Dataset sections. C.H. also helped with literature review and wrote the code for the image preprocessing. C.H. also did most of the references.

S.K. wrote the majority of the model architecture and trained and evaluated the model. S.K. also helped review the manuscript and presentation, contributed to the literature review, and wrote the majority of the Architecture and Training sections.

Acknowledgments

We would like to thank our Instructor, S. Shailja, and Teaching Assistants, Satish Kumar and Arthur Caetano, for continually providing us with ideas and feedback on our project.

We would also like to thank Summer Research Academies (SRA) and University of California Santa Barbara (UCSB) for providing the resources to make our research possible.

References

Finch, Alina, et al. "Advances in Research of Adult Gliomas." *International Journal of Molecular Sciences*, vol. 22, no. 2, Jan. 2021, p. 924. Crossref, <https://doi.org/10.3390/ijms22020924>.

Lynn M Fletcher-Heath, Lawrence O Hall, Dmitry B Goldgof, F.Reed Murtagh, Automatic segmentation of non-enhancing brain tumors in magnetic resonance images, *Artificial Intelligence in Medicine*, Volume 21, Issues 1–3, 2001, Pages 43-63, ISSN 0933-3657, [https://doi.org/10.1016/S0933-3657\(00\)00073-7](https://doi.org/10.1016/S0933-3657(00)00073-7).
(<https://www.sciencedirect.com/science/article/pii/S0933365700000737>)

Mohammad Havaei, Axel Davy, David Warde-Farley, Antoine Biard, Aaron Courville, Yoshua Bengio, Chris Pal, Pierre-Marc Jodoin, Hugo Larochelle, Brain tumor segmentation with Deep Neural Networks, *Medical Image Analysis*, Volume 35, 2017, Pages 18-31, ISSN 1361-8415,
<https://doi.org/10.1016/j.media.2016.05.004>.
(<https://www.sciencedirect.com/science/article/pii/S1361841516300330>)

Agrawal, Pulkit, et al. "Convolutional Neural Networks Mimic the Hierarchy of Visual Representations in the Human Brain." https://people.csail.mit.edu/pulkitag/data/cnn_mimics_brain.pdf

Ronneberger, O., Fischer, P., Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, N., Hornegger, J., Wells, W., Frangi, A. (eds) *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. MICCAI 2015. Lecture Notes in Computer Science(), vol 9351. Springer, Cham. https://doi.org/10.1007/978-3-319-24574-4_28

Dong, H., Yang, G., Liu, F., Mo, Y., Guo, Y. (2017). Automatic Brain Tumor Detection and Segmentation Using U-Net Based Fully Convolutional Networks. In: Valdés Hernández, M., González-Castro, V. (eds)

Medical Image Understanding and Analysis. MIUA 2017. Communications in Computer and Information Science, vol 723. Springer, Cham. https://doi.org/10.1007/978-3-319-60964-5_44

Rehman, Mobeen Ur, et al. "BU-Net: Brain Tumor Segmentation Using Modified U-Net Architecture." Electronics, vol. 9, no. 12, Dec. 2020, p. 2203. Crossref, <https://doi.org/10.3390/electronics9122203>.

Ali, Owais, et al. "Implementation of a Modified U-Net for Medical Image Segmentation on Edge Devices." IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 69, no. 11, 2022, pp. 4593–4597, <https://doi.org/10.1109/tcsii.2022.3181132>.

Dataset:

U.Baid, et al., "The RSNA-ASNR-MICCAI BraTS 2021 Benchmark on Brain Tumor Segmentation and Radiogenomic Classification", arXiv:2107.02314, 2021.

B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, 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

S. Bakas, H. Akbari, A. Sotiras, M. Bilello, M. Rozycki, J.S. Kirby, et al., "Advancing The Cancer Genome Atlas glioma MRI collections with expert segmentation labels and radiomic features", Nature Scientific Data, 4:170117 (2017) DOI: 10.1038/sdata.2017.117

Imaging Software (ITK-Snap):

Paul A. Yushkevich, Joseph Piven, Heather Cody Hazlett, Rachel Gimpel Smith, Sean Ho, James C. Gee, and Guido Gerig. User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. Neuroimage 2006 Jul 1;31(3):1116-28.

Figures:

[1] Paul A. Yushkevich, Joseph Piven, Heather Cody Hazlett, Rachel Gimpel Smith, Sean Ho, James C. Gee, and Guido Gerig. User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. Neuroimage 2006 Jul 1;31(3):1116-28.

[2] Ronneberger, O., Fischer, P., Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, N., Hornegger, J., Wells, W., Frangi, A. (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. MICCAI 2015. Lecture Notes in Computer Science(), vol 9351. Springer, Cham. https://doi.org/10.1007/978-3-319-24574-4_28

[3] Colman, Jordan, et al. "DR-Unet104 for Multimodal MRI brain tumor segmentation." Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 6th International Workshop, BrainLes 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4, 2020, Revised Selected Papers, Part II 6. Springer International Publishing, 2021.

[4] Walsh, Jason, et al. "Using U-Net Network for Efficient Brain Tumor Segmentation in MRI Images." Healthcare Analytics, vol. 2, 2 Nov. 2022, p. 100098, <https://doi.org/10.1016/j.health.2022.100098>.

[5] Rehman, Mobeen Ur, et al. "BU-Net: Brain Tumor Segmentation Using Modified U-Net Architecture." Electronics, vol. 9, no. 12, Dec. 2020, p. 2203. Crossref, <https://doi.org/10.3390/electronics9122203>.