# Medical Image Analysis using UNet

Louis Etienne Kusno 2025403026

## 1 Introduction

Medical image analysis is a critical field in healthcare. In this project the goal was to use a neural network to detect regions of legion in brain MRI images.

A UNet architecture was used for this task. UNet is a convolutional neural network that was originally designed for biomedical image segmentation [1]. It has a U-shaped architecture that consists of a contracting path to capture context and a symmetric expanding path that enables precise localization.

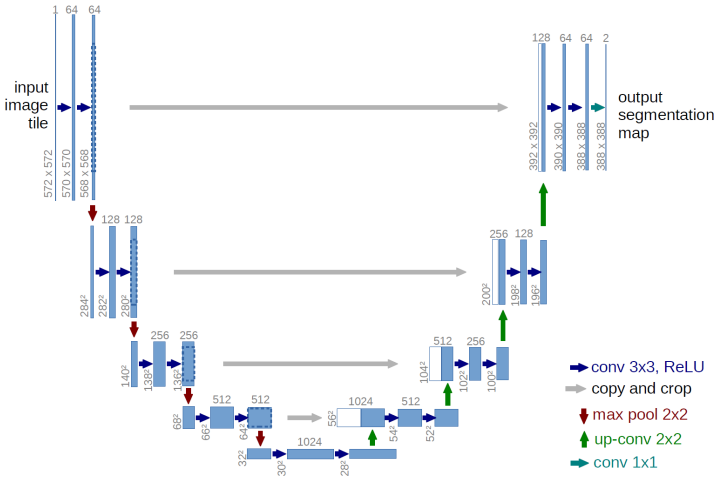


Figure 1: UNet Architecture

## 2 Architecture

The UNet architecture consists of two main parts: the encoder and the decoder. The encoder captures the context of the image, while the decoder enables precise localization. The architecture is symmetric, with skip connections that allow for better gradient flow and feature reuse.

### 2.1 Dice Coeeficient

The loss function used is the Dice coefficient, which is a measure of overlap between two samples. It is defined as:

$$\text{Dice}(A, B) = \frac{2|A \cap B|}{|A| + |B|} \tag{1}$$

where $A$ and $B$ are the predicted and ground truth masks, respectively. The Dice coefficient ranges from 0 to 1, where 1 indicates perfect overlap.

However, the Dice coefficient was also used to calculate the accuracy of the model. The accuracy is defined as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

where $TP$ is true positive, $TN$ is true negative, $FP$ is false positive, and $FN$ is false negative.

Accuracy includes true negatives (correctly labeled background pixels), which often dominate in medical images (e.g., brain scans with small lesions). As a result, a model that labels every pixel as background can achieve high accuracy while completely missing lesions.

Dice ignores true negatives, focusing only on the overlap of the foreground object. This makes it much more sensitive to how well the model segments the structure of interest, especially when that structure occupies a small fraction of the image.

Therefore, the Dice coefficient is a better metric for evaluating the performance of the model in this case.

## 3 First Training Loop

| Parameter | Value |
|---|---|
| batch_size | 16 |
| epochs | 7 |
| lr | 0.0003 |
| workers | 0 |
| image_size | 256 |
| aug_scale | 0.05 |
| aug_angle | 15 |

Table 1: Training 1 Configuration Parameters

Running the training script takes a long time, arount 40 minutes per epoch. Therefore I limited the number of epochs to 7. As we can see from Figure 2, the loss plateaus after 800 steps (around 4 epochs)

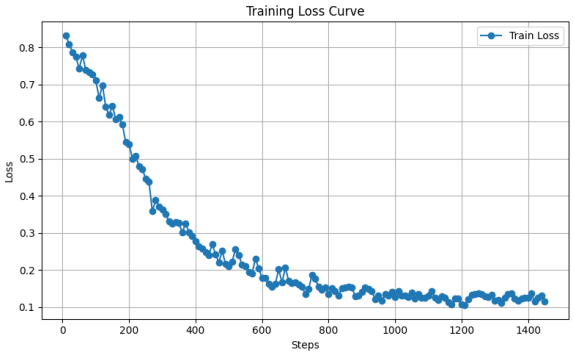


Figure 2: Loss over epochs (1)

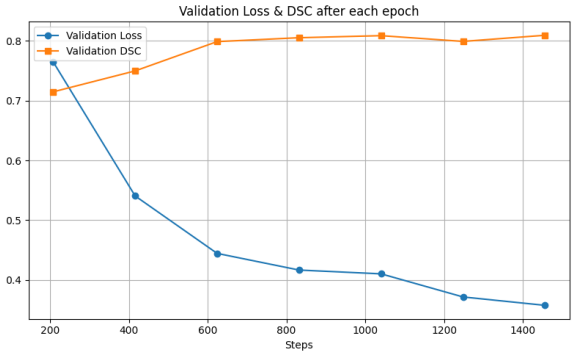


Figure 3: Validation metrics per epoch (1)

Moreover, Figure 3 shows the validation loss and DSC per epoch. This gives a better understanding of the model performance. We can see that the validation loss decreases much like over training, but its value is much higher than the training loss. This indicates that the model is overfitting to the training data. On the other hand, the validation DSC increases slightly in the beginning, but then fluctuates around 0.8.

To further evaluate the model, Figure 4 and Figure 5 show accuracy and DSC metrics for the validation set. Each results are grouped by patients. The accuracy is greater than 0.98, which is very high. However, the DSC is only around 0.8. The red line represents the mean and the green line the median.

The patient *3m\TCGA_CS_6668* has a DSC score of 0, this is because the model predicted false positives in some images and could not identify the legion in other images.

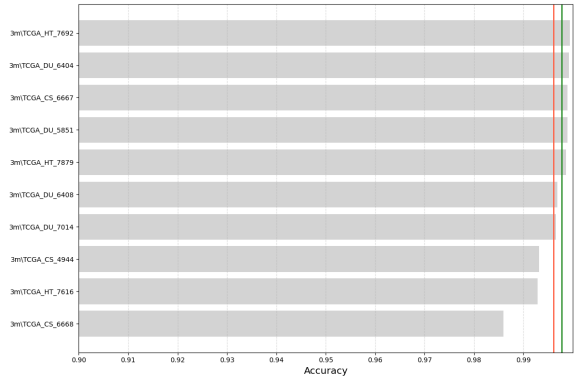


Figure 4: Validation Accuracy

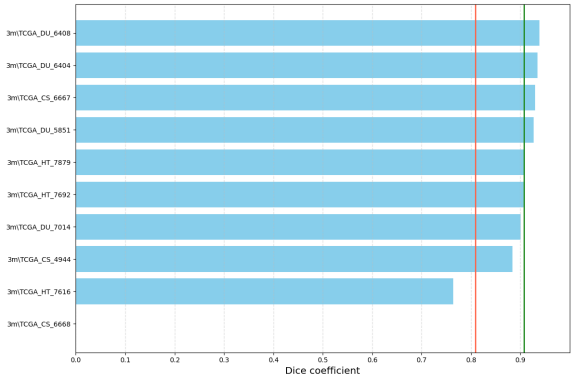


Figure 5: Validation DSC

# 4 Second Training Loop

| Parameter | Value |
|---|---|
| batch_size | 32 |
| epochs | 10 |
| lr | 0.001 |
| wd | 0.0001 |
| workers | 0 |
| image_size | 256 |
| aug_scale | 0.05 |
| aug_angle | 15 |

Table 2: Training 2 Configuration Parameters

For the second training lopp, I increased the batch size to 32 and the learning rate to 0.001. I also added weight decay to prevent overfitting. Training also took 40 minutes per epoch, so I limited it to 10 epochs this time.

The Figure 6 shows that a higher learning rate leads to a faster convergence. The loss decreases much faster than in the first training loop. However, the loss plateaus at around 0.17 instead of 0.12. This indicates that the model is not able to learn as well as in the first training loop.

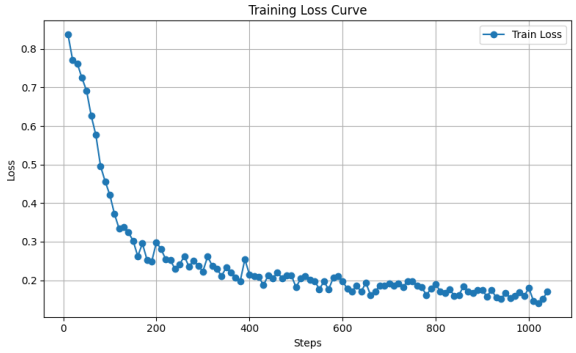


Figure 6: Loss over epochs (2)

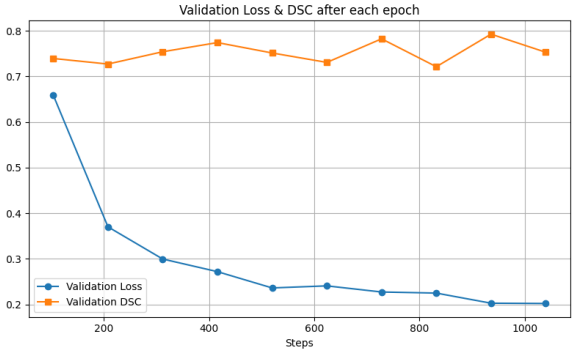


Figure 7: Validation metrics per epoch (2)

Next, here are the validation metrics per epoch. Compared to the first training loop, everything is slightly worse. For example, the mean DSC went from 0.8093 to 0.7925.

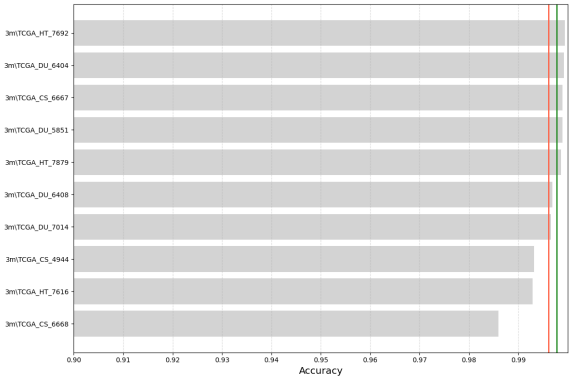


Figure 8: Validation Accuracy (2)

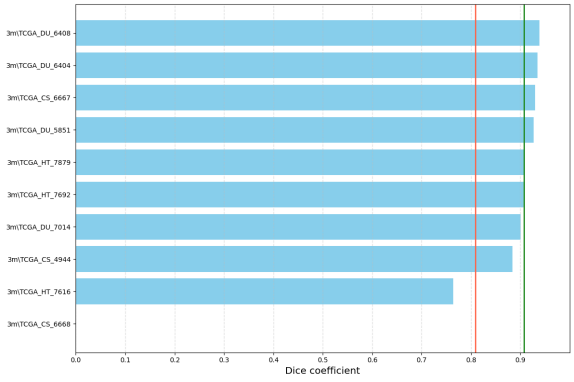


Figure 9: Validation DSC (2)

## 5 Comparison

The hyperparameters used in the first training loop were better than the second. And since the training time was very high, I changed many of them but now I am unable to see which one helped and which one worsened it.

## 6 Results

Here are some of the predictions made by the model, in red is the prediction and in green the ground truth. Both Figure 10 and Figure 11 are from the *3m\TCGA_DU_6408* patient, the one with the highest DSC score. We can see that the model is able to predict the legion quite well.

And both Figure 12 and Figure 13 are from the *3m\TCGA_CS_6668* patient, the one with the lowest DSC score. We can see that the model missed an area of legion and mistook the skull in upper sections of the brain for a legion.

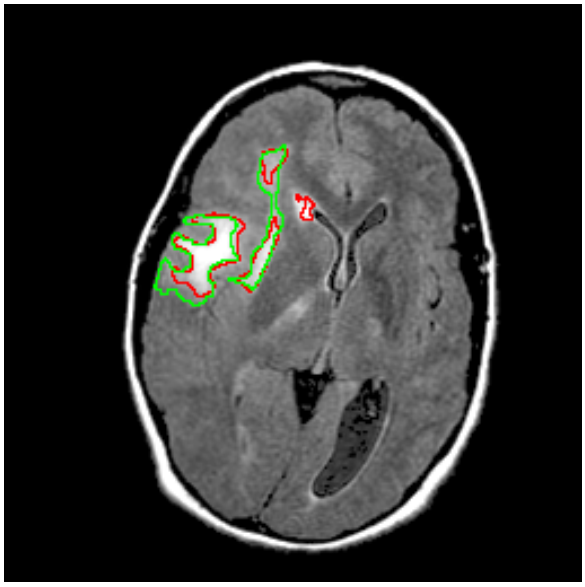

Figure 10: Prediction 1

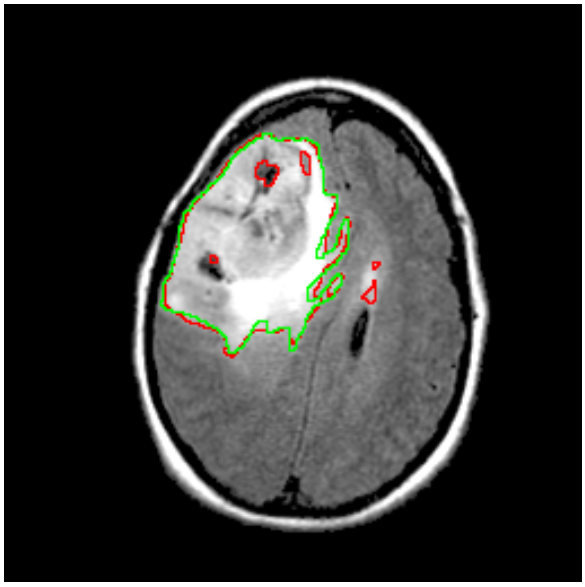

Figure 11: Prediction 2

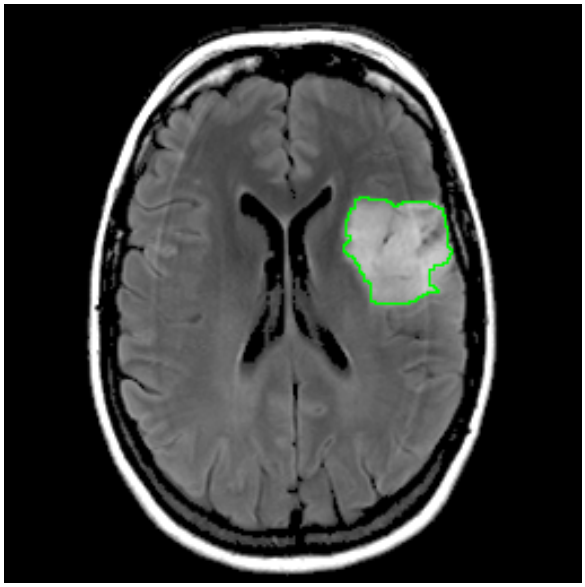

Figure 12: Prediction 3

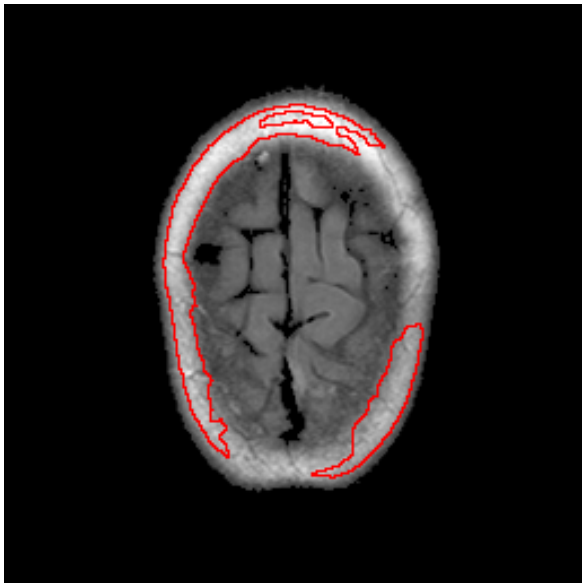

Figure 13: Prediction 4

## References


[1] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. `https://arxiv.org/abs/1505.04597`

[2] Kaggle. (n.d.). Brain MRI dataset. Retrieved from `https://www.kaggle.com/mateuszbuda/lgg-mri-segmentation`