Brain MRI Segmentation and Cancer Detection

Reza Dalir

Artificial Intelligence course Final Project, fall 2024

Project Purpose

This project focuses on brain tumor segmentation using U-Net model. The main goal is to implement this model and automatically detect and segment brain tumors from MRI scans.

Brain tumor detection is important because early diagnosis can help doctors provide better treatment. Manual segmentation by doctors is time-consuming and challenging, so we use a U-Net-based deep learning model to make the process faster and more accurate.

Dataset Overview

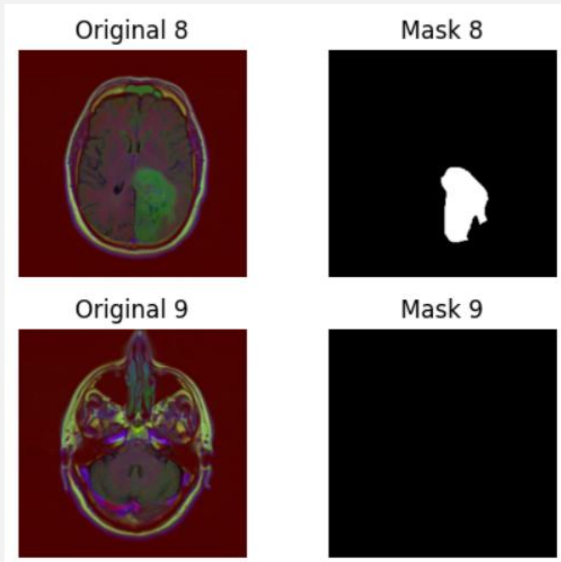
The dataset used in this project consists of brain MRI scans with corresponding segmentation masks that indicate the tumor regions. Each sample includes both MRI images which is a grayscale scan of the brain and the segmentation mask which works as the label and shows where tumors are located.

The dataset is used to train a U-Net model to automatically detect and segment tumors. It is essential for developing an AI system that can assist doctors in analyzing brain scans quickly and accurately.

Project Phases

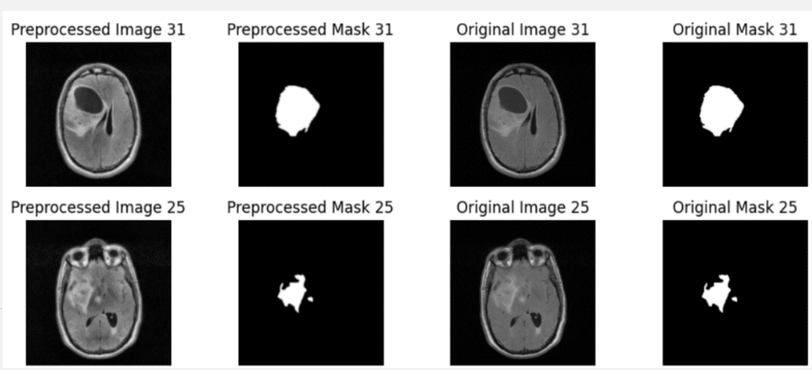
1- Dataset Familiarization

In this phase I have opened the images and their corresponding masks and plot some of them to get more familiar with the dataset. After that I split the train, validation and test sets for using in training and evaluating the model properly. Here are some examples of the dataset images and their corresponding masks, first one contains a tumor and the second one is tumor-free:



2- Preprocessing

In this phase I preprocessed the images and their corresponding masks with the use of different preprocessing methods. At first, I converted the images to gray scale and resized each one to 256*256 to avoid collision. Then I normalized each image to the range [0, 1] and finally I applied CLAHE and a gaussian blur mask for maintaining the contrast of the image. Here is a comparison between some preprocessed and original images:



3- Implementation, Training, and Validation of the Model

I have implemented the U-Net model in a function, the structure of this model contains several parts:

- Encoder: A series of convolutional layers with ReLU activation, followed by max pooling to capture features at different levels.
- Bottleneck: The deepest layer of the network that captures the most important tumor features.
- Decoder (Expanding Path): Upsampling layers to reconstruct the segmented tumor areas, with skip connections to retain fine details.
- Output Layer: A 1×1 convolution with a sigmoid activation to generate the final binary segmentation mask (tumor vs. background).

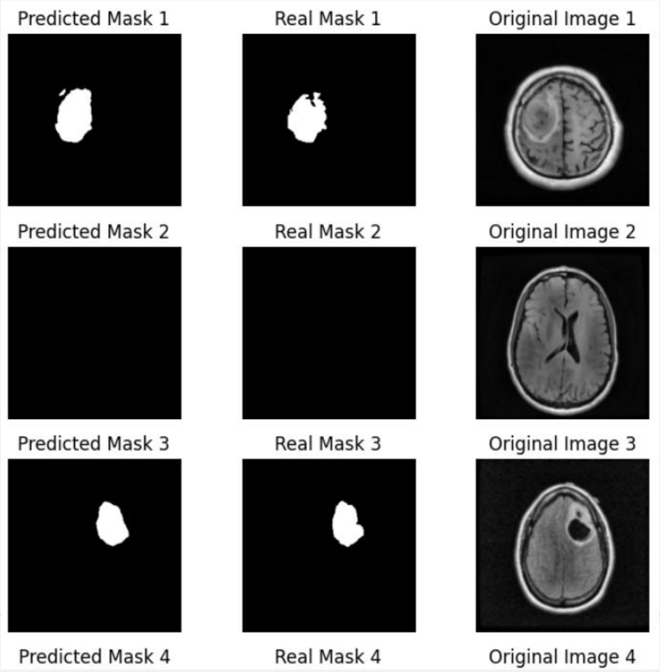
At first, I used Dice Loss + Binary Cross entropy as loss function and trained the model with 15 epochs, here is the evaluation results and a sample plot on the test set:

IoU: 0.5626133903178764

Dice Coefficient: 0.7200928826079317

Rand Error: 0.005419731140136719

Pixel Error: 0.005419704446962468



As we can see in plots, predicted masks are almost close to the actual masks and metrices shown almost a good result.

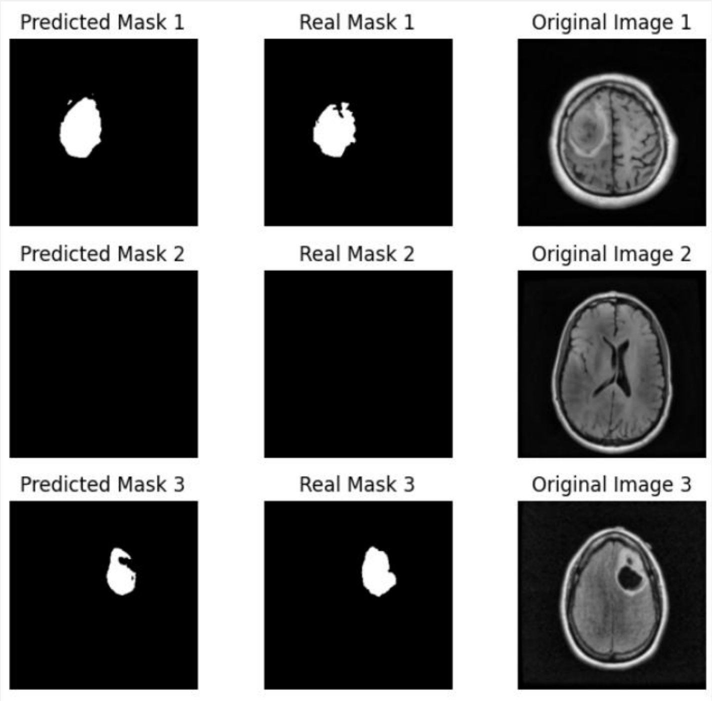
Then I tried binary cross entropy as loss function and trained the dataset again with the same other parameters, here are the results as well:

IoU: 0.6780564243413943

Dice Coefficient: 0.8081449640258894

Rand Error: 0.003940582275390625

Pixel Error: 0.003940650221652354



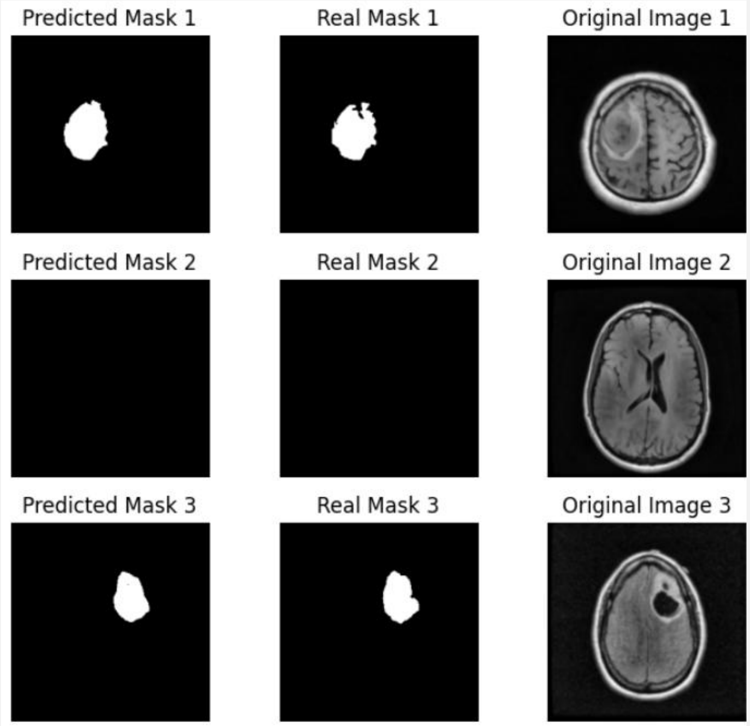
As we can see metrices show that performance is now better and the plot shows that as well. So, I used the model with these parameters and done the training phase again with 50 epochs, but early stopping condition stopped the training after 41 epochs. And here is the final evaluation on test set with the U-Net model:

IoU: 0.7734004739311859

Dice Coefficient: 0.8722231501571106

Rand Error: 0.0027622580528259277

Pixel Error: 0.0027623067375357824



The model achieved an IoU score of 0.7734 and a Dice coefficient of 0.8722, indicating high segmentation accuracy. IoU measures the overlap between the predicted and actual segmentation masks. It is calculated as the intersection of the two masks divided by their union. Higher values indicate better segmentation.
Dice coefficient is similar to IoU, but it emphasizes overlap more. It is calculated as (2 × intersection) / (sum of both masks). A higher Dice score means better segmentation performance.
The low Rand Error (**0.0027**) and Pixel Error (**0.0027**) further confirm that the model makes very few misclassifications.
The visual comparison between predicted masks and real masks shows that the model effectively captures tumor regions, although slight differences exist. The

model performs well, but minor improvements in boundary detection could further enhance its accuracy.

Overly, the U-Net-based segmentation model provides strong performance in brain tumor segmentation, making it a promising tool for medical image analysis.

In the main article rand error and pixel error are around **0.0382** and **0.0611** respectively. So, comparing the result with the original method used in the article our model performance was better since we used preprocessing and some other methods.

During the project, I faced several challenges, including handling the high memory consumption of certain evaluation metrics like the Rand Error, which caused performance issues. Training the U-Net model required careful tuning of hyperparameters to prevent overfitting while ensuring accurate segmentation. Another challenge was preprocessing MRI images effectively, such as applying CLAHE for contrast enhancement and normalizing pixel values. Additionally, balancing the dataset and dealing with class imbalance in tumor segmentation were key difficulties in achieving reliable predictions.

Bonus part:

For the bonus part, I modified the U-Net model by adding an attention mechanism. This improves how the model focuses on important areas in brain MRI scans. I added attention gates in the skip connections of the U-Net. These gates help the model learn which parts of the image are most important for segmentation, reducing unnecessary background noise. Normally, the U-Net directly connects encoder layers to decoder layers. In my version, I added attention layers that filter out irrelevant information before passing features to the decoder. The attention mechanism helps the model concentrate on tumor regions while ignoring less relevant areas. This improves accuracy, especially in complex cases where tumors are small or unclear.

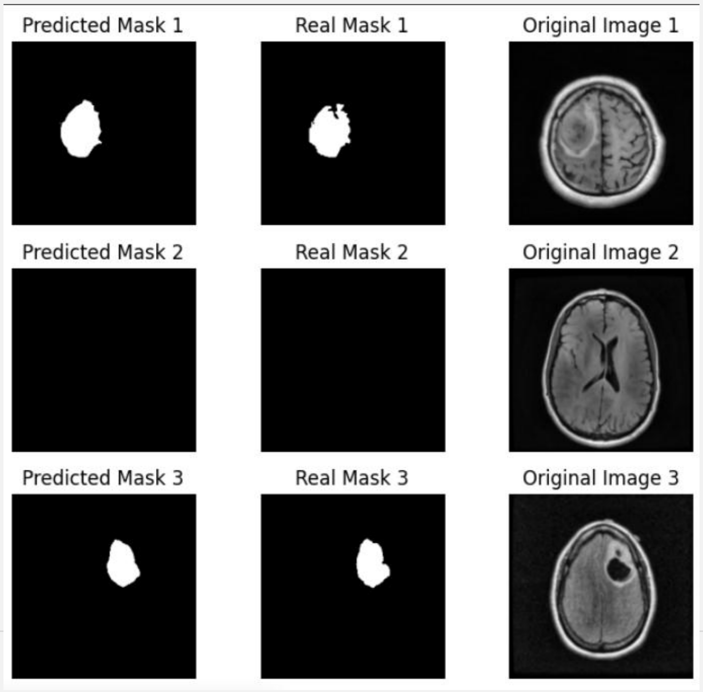
By adding attention, my U-Net model can make more precise predictions and improve segmentation performance. Here is the performance with the same parameters and 50 epochs:

IoU: 0.761609419013652

Dice Coefficient: 0.8646745536136915

Rand Error: 0.0031114816665649414

Pixel Error: 0.0031115505228212466



Even though the accuracy is a bit less than the original U-Net model but it can locate the boundaries better than the original U-Net model. So in practice it works better for this task.