University Abdelmalek Essaâdi

National School of Applied Sciences of Tétouan

Second Year of the Engineering Cycle – BDIA

Module: Deep Learning

Brain Tumor Segmentation

Authors:

BOUSKINE OTHMANE

BOULAALAM YASSINE

Supervisor:

Prof. BELCAID ANASS

Academic Year: 2024/2025

Contents

I. Introduction

Brain tumor segmentation is a critical task in medical imaging that involves the precise delineation of tumor regions in brain scans, primarily using Magnetic Resonance Imaging (MRI). Accurate segmentation is essential for diagnosis, treatment planning, and monitoring disease progression, as it allows clinicians to better understand tumor morphology and its impact on surrounding structures.

Deep learning techniques, particularly convolutional neural networks (CNNs), have greatly advanced the field of medical image analysis, offering robust solutions to complex challenges. In brain tumor segmentation, achieving reliable results remains challenging due to the variability in tumor size, shape, and intensity patterns in MRI scans. Architectures such as U-Net and its derivatives are widely recognized for their effectiveness in capturing both spatial details and contextual information, making them well-suited for biomedical image segmentation tasks.

This report explores brain tumor segmentation using two architectures: the standard U-Net and a variant utilizing VGG16 as its encoder. It also evaluates the performance of these models using specific loss functions such as BCE-Dice and Tversky, which are designed to address the unique challenges of medical image segmentation, including imbalanced data. The evaluation metrics and findings presented aim to highlight the potential and limitations of these techniques in real-world applications.

II. Dataset

The dataset used in this project consists of 3,929 MRI scans and their corresponding 3,929 binary segmentation masks, designed specifically for brain tumor segmentation. It includes data from approximately 110 patients, with each patient contributing multiple MRI slices, providing a comprehensive representation of brain anatomy. Each image in the dataset is accompanied by binary segmentation masks that serve as annotations, explicitly highlighting tumor regions within the scans. These annotations are critical for training and validating the segmentation model, ensuring it learns to differentiate tumor regions from healthy brain tissue accurately. The images are stored in the .tif format, a high-quality format suitable for preserving the intricate details of medical imaging. This dataset's rich annotations and diverse patient data make it well-suited for developing and evaluating robust segmentation models, enabling precise identification of brain tumor regions across varying MRI slices.

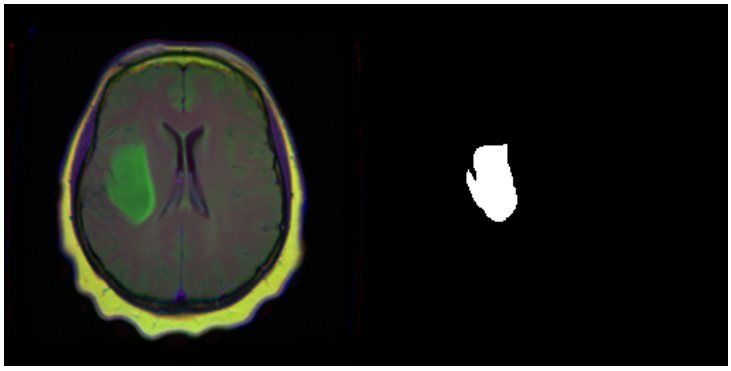


Figure II.1: Sample from the LGG dataset of MRI scan and its corresponding binary segmentation mask. https://www.kaggle.com/datasets/mateuszbuda/lgg-mri-segmentation

III. Architecture

III.1 Models

III.1.1 U-Net

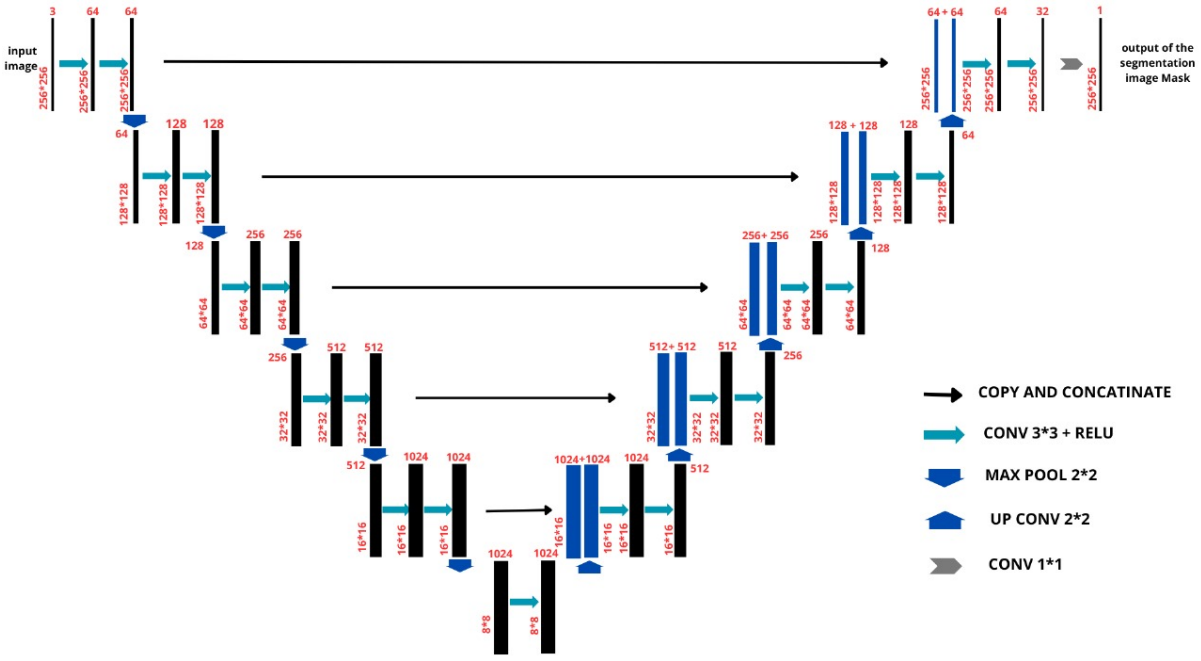


Figure III.1: The architecture of the U-Net model used for brain tumor segmentation.

The U-Net architecture consists of three main components: the encoder, bottleneck, and decoder. The encoder reduces the input image size from 256×256×3 to 8×8×1024 through five downsampling blocks. Each block includes 3×3 convolutions to retain spatial resolution, batch normalization for training stability, ReLU activation for non-linear pattern learning, and 2×2 max-pooling to reduce spatial dimensions while capturing high-level tumor-specific features. Skip connections between the encoder and decoder preserve critical spatial details necessary for accurate tumor boundary delineation.

At the center of the architecture, the bottleneck processes features at the smallest spatial resolution (8×8×1024) using 3×3 convolutions to refine and capture deep, abstract tumor representations. This enables the model to identify subtle variations in tumor intensity and shape, crucial for precise segmentation.

The decoder reconstructs the segmentation map by progressively upsampling the feature maps back to the original image dimensions (256×256). It uses five upsampling blocks with 2×2 transposed convolutions, followed by 3×3 convolutions. Skip connections integrate encoder features with upsampled decoder features, blending high-level tumor features with fine spatial details. A final 1×1 convolution reduces the feature map's channels to one, generating a binary segmentation mask with pixel-wise tumor probabilities. This ensures both the tumor boundaries and internal structure are accurately captured.

III.1.2 U-Net-VGG16

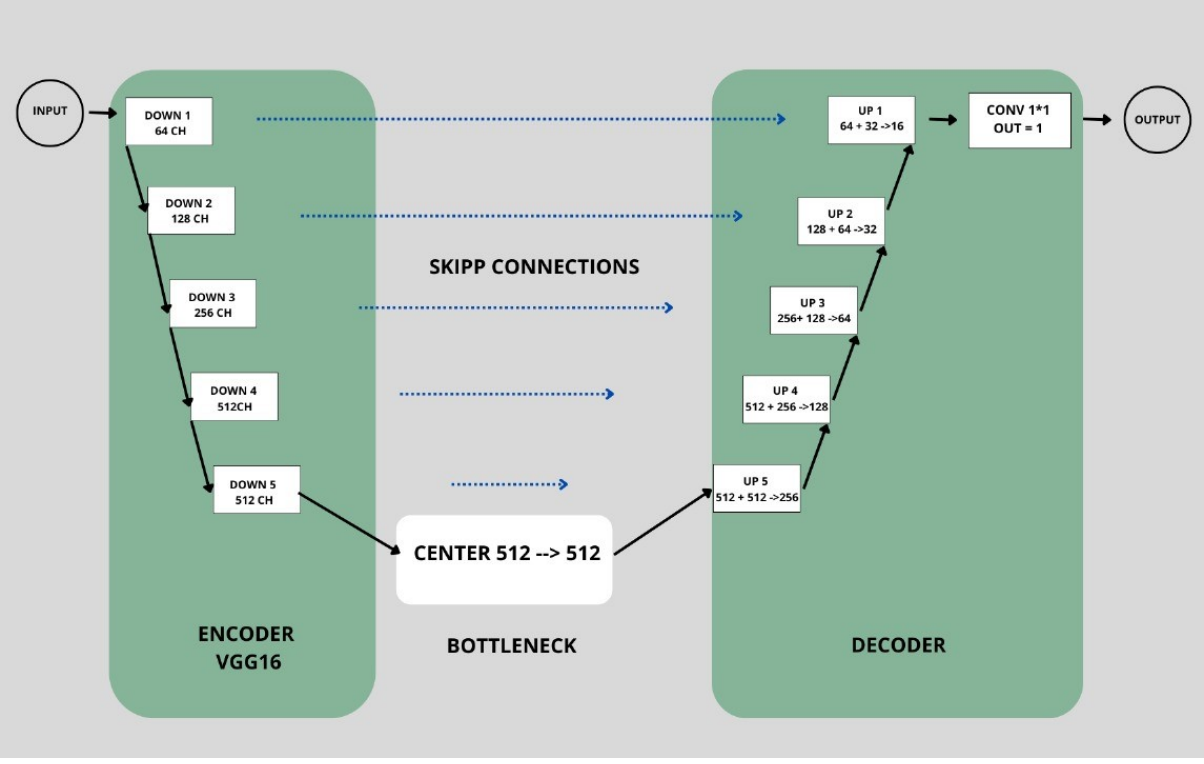


Figure III.2: The architecture of the U-Net-VGG16 model used for brain tumor segmentation.

The U-Net architecture with VGG16 as its encoder is specifically designed for brain tumor segmentation, combining robust feature extraction with precise reconstruction.

The encoder, derived from VGG16, processes MRI scans through five blocks, progressively reducing spatial resolution while capturing essential tumor features such as irregular shapes and intensity variations. The bottleneck further refines these features, emphasizing critical tumor-specific information.

The decoder reconstructs detailed segmentation masks by progressively upsampling feature maps using transposed convolutions. Skip connections integrate high-level semantic information from the encoder with low-level spatial details, ensuring accurate tumor boundary delineation. The final layer applies a 1×1 convolution and a sigmoid activation function to generate pixel-wise probabilities, creating binary segmentation masks. This design, leveraging VGG16's strengths and a custom decoder, is optimized for the precise and context-aware segmentation of brain tumors.

III.2 Model Configuration

The logic adopted in this work is centered on building a robust and reliable approach tailored for brain tumor identification in MRI scans. The methodology ensures that the model effectively learns from the available data and generalizes well to unseen cases. This involves key steps such as augmenting the dataset to simulate diverse conditions, splitting the data into training and testing sets, and employing optimization techniques to enhance model convergence. Furthermore, specialized loss functions are used to address the challenges of imbalanced data and improve segmentation accuracy. Each component of this methodology is designed to ensure the development of a highly accurate and efficient segmentation model.

III.2.1 Data Augmentation

Data augmentation is a crucial preprocessing step aimed at enhancing the variability and diversity of the training dataset, thereby improving the model's ability to generalize to unseen data. The augmentation process is implemented using the Albumentations library, focusing on key transformations. For the training dataset, all images are resized to a standardized resolution of 256×256 pixels. Augmentation steps include horizontal and vertical flipping, random 90-degree rotations, and shift, scale, and rotation adjustments

to introduce spatial variability. Additional enhancements involve random brightness and contrast modifications to simulate changes in imaging conditions and the application of Gaussian noise to mimic potential artifacts in medical imaging. These transformations collectively produce a diverse set of training samples, increasing the model's robustness. For the testing dataset, resizing is the only transformation applied, ensuring consistency in input size while maintaining the original features for accurate evaluation.

III.2.2 Data Splitting

the dataset is strategically split into training and testing subsets to ensure effective model training and evaluation. The dataset is first loaded into a custom BrainDataset object, which serves as a centralized structure for managing image data and corresponding annotations. The dataset is then divided, with 80% allocated to the training set and the remaining 20% reserved for testing. This ratio is chosen to provide the model with sufficient data for learning while maintaining a significant portion for unbiased evaluation. After splitting, appropriate transformations are applied to each subset. The training dataset undergoes a series of augmentations to introduce variability and improve generalization, while the testing dataset is resized without further alterations to preserve its original characteristics for evaluation.

DataLoaders are subsequently created for both subsets, with the training DataLoader configured to shuffle the data to reduce potential bias and the testing DataLoader maintaining a fixed order for consistency.

III.2.3 Optimization Function

We chose the Adam optimizer for this project due to its efficiency and adaptability in training deep learning models, especially for complex tasks like brain tumor segmentation. Adam adjusts learning rates individually for each parameter based on the mean and variance of gradients, enabling stable updates despite the variability inherent in medical imaging data. With a learning rate of 10^{-4}, it ensures steady convergence while reducing the risk of overshooting optimal parameters. This adaptability allows the optimizer to navigate the challenging loss landscape effectively, capturing intricate tumor details without being overly influenced by noise. The balance between precision and robustness

makes Adam an ideal choice for achieving high segmentation accuracy.

III.2.4 Scheduler

The ReduceLROnPlateau learning rate scheduler was used to reduce the learning rate when the model's performance plateaus. With a patience value of 3, the learning rate is decreased after three epochs without improvement in the monitored metric. This helps ensure stable and efficient convergence by adjusting the learning rate dynamically during training.

III.2.5 Loss Functions

Loss functions are critical in guiding the model's training process, as they quantify the difference between the predicted output and the ground truth. For this project, two advanced loss functions were utilized to optimize the segmentation performance: BCE-DICE Loss and Tversky Loss.

Binary Cross-Entropy + Dice Loss

The Binary Cross-Entropy + Dice Loss combines two complementary loss functions: Binary Cross-Entropy (BCE) and Dice loss. The formula is as follows:

$$\mathcal{L}_{BCE+DICE} = \mathcal{L}_{BCE} + \mathcal{L}_{DICE}$$

The Binary Cross-Entropy (BCE) loss is defined as:

$$\mathcal{L}_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}\Big(y_i \log(p_i) + (1-y_i)\log(1-p_i)\Big)$$

The Dice loss is derived from the Dice Score:

$$\mathcal{L}_{DICE} = 1 - DiceScore$$

The Dice Score is defined as:

$$DiceScore = \frac{2|A \cap B|}{|A| + |B|}$$

Where:

- A: Predicted segmentation region (p_i).
- B: Ground truth segmentation region (y_i).
- $|A \cap B|$: The area of overlap between A and B.
- $|A|, |B|$: The total areas of the predicted and ground truth regions, respectively.

Thus, the full BCE + Dice Loss becomes:

$$\mathcal{L}_{BCE+DICE} = -\frac{1}{N}\sum_{i=1}^{N}\Big(y_i \log(p_i) + (1-y_i)\log(1-p_i)\Big) + 1 - \frac{2|A \cap B|}{|A| + |B|}$$

Where:

- N: Total number of pixels in the image.
- ϵ: A small constant (added to $|A|$ and $|B|$ to prevent division by zero, if necessary).

This hybrid loss function strikes a balance between pixel-level accuracy and overall segmentation shape similarity. The BCE component focuses on minimizing pixel-wise errors, while the DICE component ensures better overlap and consistency in the segmented regions, which is crucial for accurate brain tumor segmentation where both fine boundaries and global shape are important.

Tversky Loss

The formula for the Tversky Loss is given by:

$$\mathcal{L}_{Tversky} = 1 - \frac{TP}{TP + \alpha \cdot FN + \beta \cdot FP}$$

Where:

- TP: The number of true positives.
- FN: The number of false negatives.
- FP: The number of false positives.

- α and β: Weighting factors for false negatives and false positives, respectively.

In this project, we chose $\alpha = 0.3$ and $\beta = 0.7$. This configuration places more emphasis on minimizing false positives, which helps ensure that the model does not mistakenly classify surrounding healthy tissue as tumor tissue. The choice of these values is motivated by the need to prioritize the detection of tumor regions, which are often smaller and more difficult to identify, while being less sensitive to minor inaccuracies in the background segmentation. The Tversky loss is well-suited for handling class imbalance in medical image segmentation tasks like brain tumor detection.

IV. Evaluation

IV.1 Metrics

The Dice Score and Tversky Index are important metrics for evaluating segmentation tasks, particularly in medical imaging. The Dice Score measures the similarity between predicted and ground truth segmentation masks by balancing precision and recall, making it effective for datasets with imbalanced classes, such as tumor segmentation. It provides a straightforward way to evaluate how well the predicted mask overlaps with the true mask. On the other hand, the Tversky Index extends the Dice Score by introducing flexibility through weighting factors that allow prioritization of false negatives or false positives. This makes it especially useful in cases where missing a tumor (false negative) is more critical than incorrectly labeling healthy tissue (false positive). While the Dice Score is a general-purpose metric, the Tversky Index is tailored for tasks with specific error priorities, providing a more nuanced evaluation for imbalanced and high-stakes segmentation problems.

IV.1.1 Dice Score

$$DiceScore = \frac{2\sum_{i=1}^{N} y_i p_i}{\sum_{i=1}^{N} y_i + \sum_{i=1}^{N} p_i}$$

Where:

- y_i: Ground truth value for pixel i (0 or 1).
- p_i: Predicted value for pixel i (typically a probability between 0 and 1).
- N: Total number of pixels in the segmentation mask.

IV.1.2 Tversky

$$TverskyIndex = \frac{TP}{TP + \alpha \cdot FN + \beta \cdot FP}$$

Where:

- TP: True positives, the number of pixels correctly predicted as part of the object.
- FN: False negatives, the number of object pixels incorrectly predicted as background.
- FP: False positives, the number of background pixels incorrectly predicted as part of the object.
- α: Weighting factor for false negatives.
- β: Weighting factor for false positives.

IV.2 Performances

Visualization of training and validation metrics.

IV.2.1 U-Net

U-Net (BCE + Dice Loss)

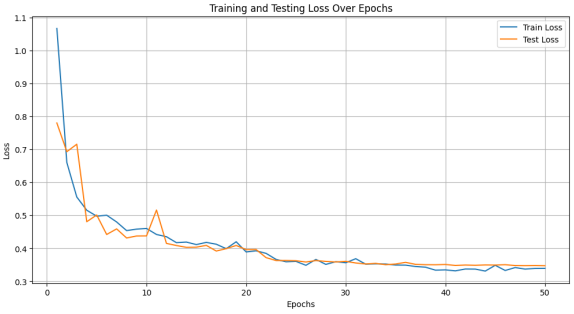


Figure IV.1: Loss in train and test for U-Net (BCE + Dice Loss).

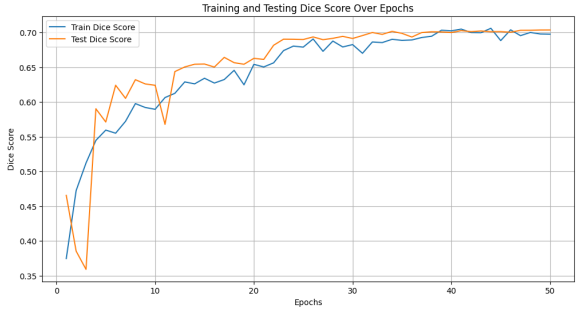


Figure IV.2: Dice score of train and test for U-Net (BCE + Dice Loss).

U-Net (Tversky Loss)

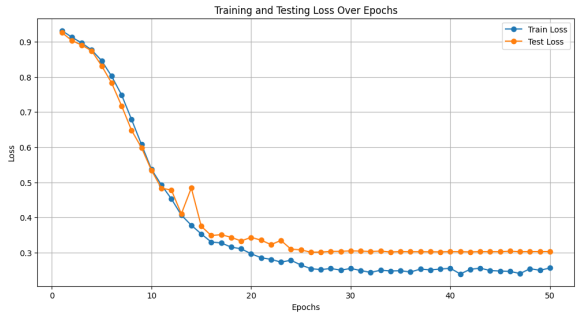


Figure IV.3: Loss of train and test for U-Net (Tversky Loss).

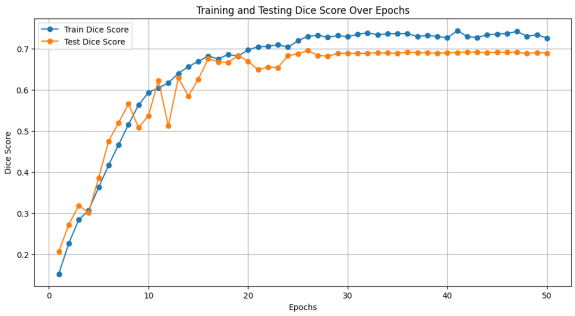


Figure IV.4: Dice score of train and test for U-Net (Tversky Loss).

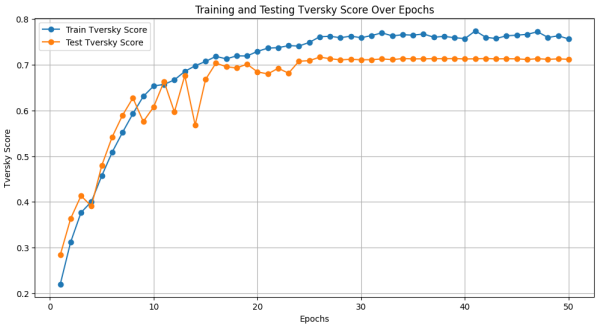


Figure IV.5: Tversky index of train and test for U-Net (Tversky Loss).

IV.2.2 U-Net-VGG16

U-Net-VGG16 (BCE + Dice Loss)

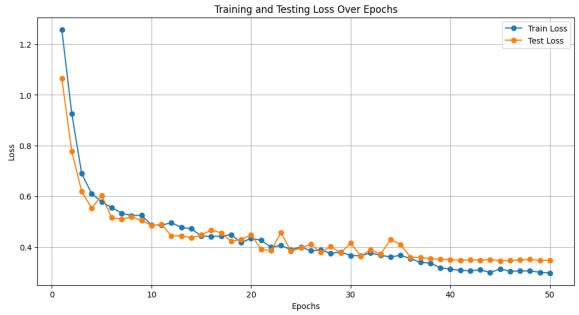


Figure IV.6: Loss of train and test for U-Net-VGG16 (BCE + Dice Loss).

Figure IV.7: Dice score of train and test for U-Net-VGG16 (BCE + Dice Loss).

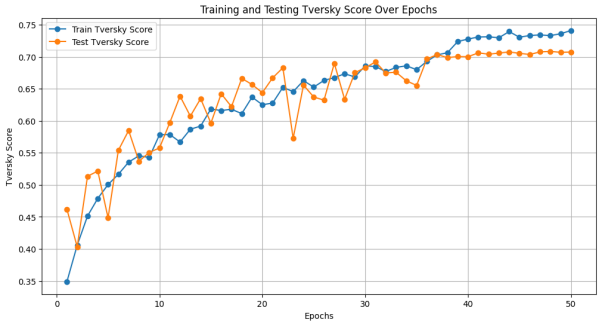


Figure IV.8: Tversky index of train and test for U-Net-VGG16 (BCE + Dice Loss).

U-Net with VGG16 (Tversky Loss)

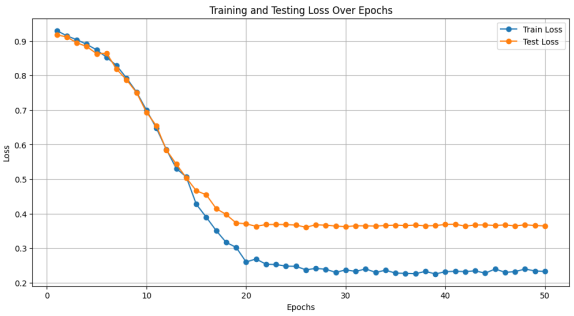


Figure IV.9: Loss of train and test for U-Net-VGG16 (Tversky Loss).

Figure IV.10: Dice score of train and test for U-Net-VGG16 (Tversky Loss).

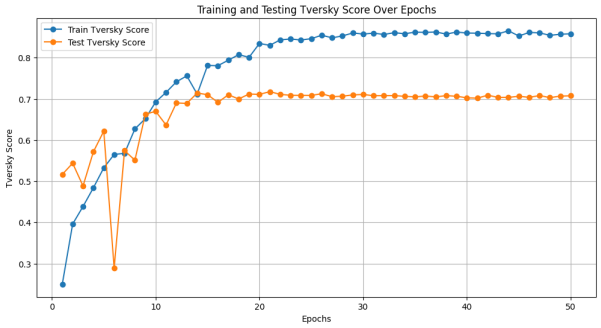


Figure IV.11: Tversky index of train and test for U-Net-VGG16 (Tversky Loss).

IV.3 Tables of Performances

Model	Loss Value	Dice-Score	Tversky-Index
Unet (BCE+Dice Loss)	0.3474	0.7034	
Unet (Tversky Loss)	0.3000	0.6959	0.7168
Unet_Vgg16 (BCE+Dice Loss)	0.3452	0.6934	0.7083
Unet_Vgg16 (Tversky Loss)	0.3634	0.6996	0.7175

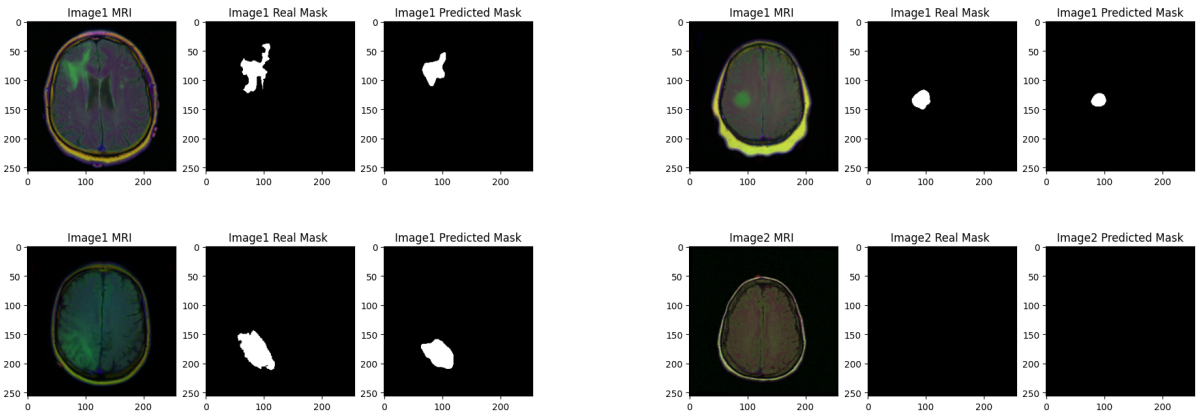
Table IV.1: Performance comparison of different models.

The table summarizes the final evaluation of the U-Net and U-Net-VGG16 architectures, tested with BCE+Dice Loss and Tversky Loss, for the task of brain tumor segmentation. The models were evaluated using Loss Value, Dice Score, and Tversky Index. The results indicate that Tversky Loss achieves better Tversky Index values, highlighting its effectiveness in handling class imbalance, which is critical in medical image segmentation tasks such as brain tumor detection. Meanwhile, BCE+Dice Loss shows slightly better performance in terms of Dice Score, suggesting its strength in optimizing region overlap.

IV.4 Samples of Results

Visual Results of Brain Tumor Segmentation

The following images present examples of MRI scans, their corresponding ground truth segmentation masks (Real Mask), and the predicted segmentation masks (Predicted Mask) generated by the model. These visualizations demonstrate the model's ability to identify tumor regions in different scenarios, highlighting both accurate predictions and areas for improvement.



V. Conclusion

In conclusion, the work presented in this study demonstrates the effectiveness of deep learning models, particularly U-Net with a VGG16 encoder, for brain tumor segmentation in MRI scans. Through careful consideration of the dataset, architecture, and training strategy, including the use of data augmentation, optimal data splitting, and the Adam optimizer, the model was able to achieve strong performance. The combination of BCE-DICE and Tversky loss functions played a critical role in addressing both pixel-level accuracy and the overall shape of tumor regions, which are essential in medical imaging tasks. The evaluation results, supported by relevant metrics such as Dice score and Tversky index, further underscore the robustness of the model in accurately segmenting brain tumors. This study highlights the potential of deep learning approaches in the medical imaging field, particularly in improving diagnosis and treatment planning for brain tumor patients.