Brain Tumor Segmentation Using U-Net Architecture

Designed by DEKDEGUE Hajar

Supervised by Pr. El Habib BENLAHMAR

Hassan II University of Casablanca - Faculty of Sciences Ben M'Sik - Department of Mathematics and Computer Science

Abstract

Manual segmentation of brain magnetic resonance imaging (MRI) scans, particularly for detecting gliomas and other anomalies, is time-consuming and prone to variability among radiologists. Automated segmentation methods, powered by deep learning, offer the potential to standardize and expedite these processes. This project focuses on developing a deep learning-based model using a U-Net architecture to automate the segmentation of brain abnormalities. The model was trained on MRI datasets from The Cancer Imaging Archive (TCIA) and BraTS 2020, comprising both 2D and 3D scans with manually annotated masks. The study addresses two key tasks: 2D binary segmentation of abnormal versus healthy tissue and 3D segmentation to capture volumetric tumor structures. Preprocessing techniques such as cropping, normalization, and augmentation were employed to enhance model performance. Initial experiments show promising results in delineating tumor boundaries, as evaluated by metrics like the Dice coefficient and Intersection over Union (IoU). This work underscores the potential of AI-driven segmentation tools to improve diagnostic efficiency and precision in neuro-oncology.

Introduction

Brain tumors, particularly gliomas, are complex and require precise segmentation in medical imaging. Manual segmentation is time-consuming and inconsistent, but deep learning models like U-Net offer a more efficient and reliable automated solution.

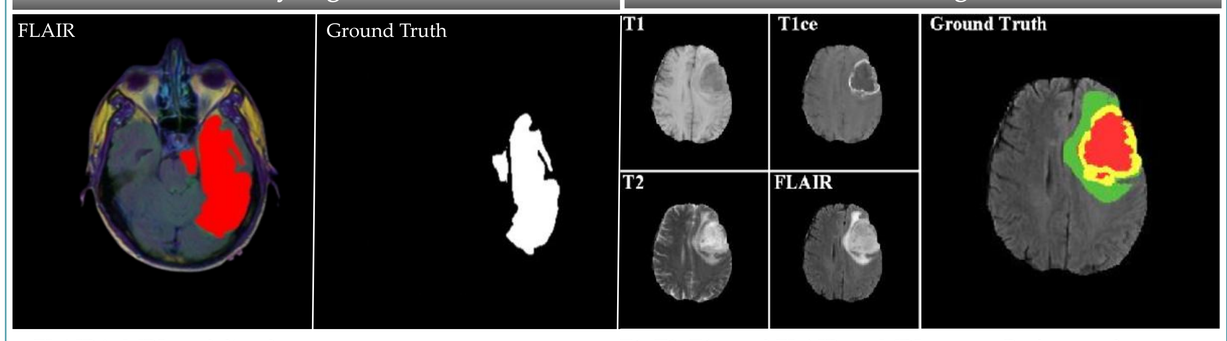
This project develops an AI-based model trained on MRI data from TCIA and BraTS 2020, focusing on 2D binary segmentation of abnormal vs. healthy tissue and 3D multiclass segmentation of tumor volumes. The goal is to enhance diagnostic accuracy and streamline clinical workflows in brain tumor treatment.

Materials & Methods

Datasets

- **The Cancer Imaging Archive (TCIA)**: Used for 2D binary segmentation. Includes brain MRI scans with manually annotated masks for abnormal vs. healthy tissue.
- **BraTS 2020**: Used for 3D multiclass segmentation. Contains MRI scans and segmentation labels for tumor subregions (enhancing tumor, non-enhancing tumor, edema).

2D Binary Segmentation



FLAIR is MRI modality that highlights much better the brain lesions better than other modalities. wen it comes to binary segmentation problems

- Background
- Tumor

The figure displays a sample MRI image alongside its annotated mask, illustrating abnormal vs. healty tissue

3D multiclass Segmentation

T1, T2, T1c, and FLAIR are MRI modalities that provide complementary information about brain tissue characteristics, enhancing the characterization of tumors and abnormalities.

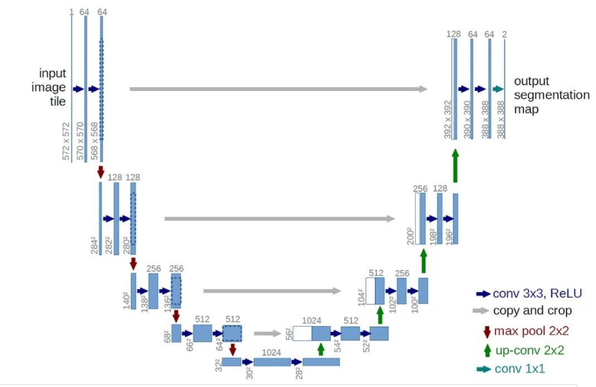
- Background
- Edema
- Non-enhancing tumor
- Enhancing tumor

The figure displays a sample MRI images of each modality alongside their corresponding mask, highlighting different tumor subregions

Preprocessing

- **2D Binary Segmentation:**
  - Resizing: All images were resized to a uniform shape of 256x256 pixels, ensuring consistency in input size across the dataset.
  - Normalization: Pixel values were normalized to scale between 0 and 1, enhancing model convergence during training.
  - Data Augmentation: Techniques such as rotation, flipping, and zooming were applied to increase the diversity of training samples and reduce overfitting.
  - Class Balance: To promote balanced learning, the dataset was adjusted to contain a comparable number of healthy and abnormal masks, mitigating class imbalance and reducing the risk of biased predictions.
- **3D Multiclass Segmentation:**
  - **Cropping** : Volumes were cropped to focus on the brain region, reducing irrelevant background and computational complexity.
  - **Modality Combination :** T2, T1c, and FLAIR modalities were merged into a single 3-channel volume (excluding T1 due to its limited contribution). Each modality provides unique insights into tumor characteristics, enhancing the model's ability to differentiate tumor subregions.
  - **Normalization :** Intensity normalization was applied on the MRI voulumes to ensure consistent pixel values across the dataset.
  - **Excluding MRI volumes with <1% Non-Background Pixels :** Only images with masks that have more than 1% useful volume (labels other than 0) will be retained, as training the model on background regions doesn't provide additional useful information.
  - **One-Hot Encoding of Masks :** converting the mask labels into a one-hot encoded format with four classes (background and the three tumor subregions), making them suitable for multiclass segmentation by the model.

Model Architecture



The figure illustrates the U-shaped architecture, which includes an encoder for feature extraction, a decoder for restoring spatial resolution, and skip connections that enhance the model's ability to combine low-level and high-level features, making it well-suited for medical image analysis.

U-Net is a convolutional neural network designed for image segmentation, particularly in medical imaging. It features an encoder to capture essential features and a decoder to restore spatial details, ensuring accurate segmentation. Skip connections facilitate the combination of detailed features from the encoder with the decoder's output, enhancing segmentation precision. U-Net is highly effective even with small datasets, making it an ideal choice for segmenting complex structures, such as tumors, in both 2D and 3D MRI scans.

Training & Evaluation

2D Binary Segmentation

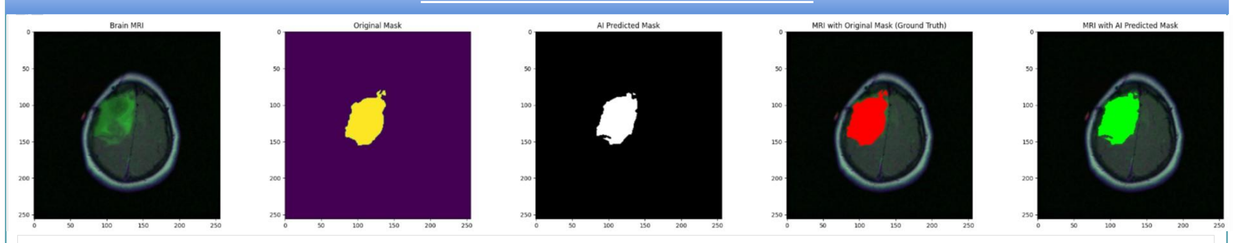
The model was trained for 100 epochs using the Adam optimizer with a learning rate of 0.001. It was compiled with the binary cross-entropy loss function and accuracy as the evaluation metric.

2D Multiclass Segmentation

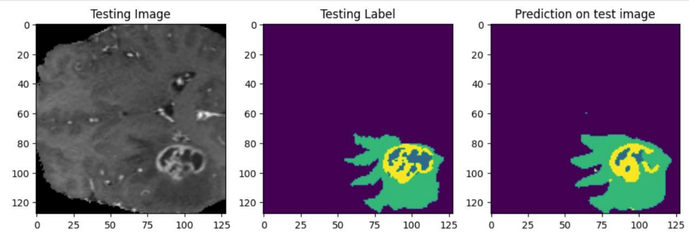
The model was trained for 100 epochs using a combination of Dice loss and Categorical Focal loss, with class weights of 0.25 for each class. The Adam optimizer was employed with a learning rate of 0.0001, and accuracy was used as the evaluation metric.

Both models were evaluated using the Dice coefficient and Intersection over Union (IoU) metrics to assess their segmentation performance.

Result & Discussion

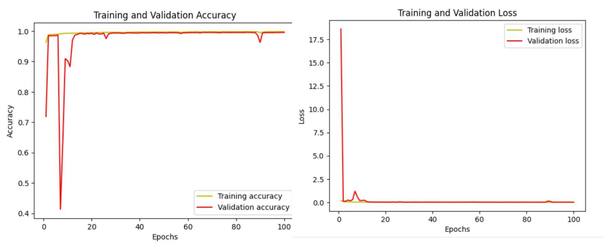


The figure above illustrates the binary segmentation performance of the model, displaying the original MRI scan, the ground truth mask, and the predicted mask. The close alignment between the predicted and real mask reflects the model's accuracy, achieving an **IoU of 0.8618**.



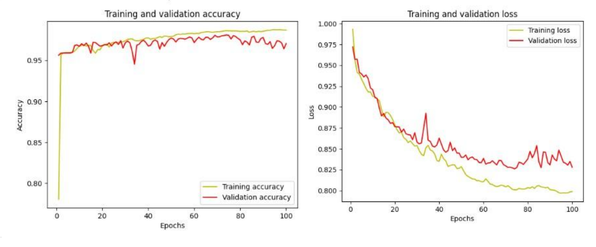
The figure displays a 2D slice from a 3D MRI volume with ground truth and predicted segmentations for multiple tumor subregions. The model effectively captures the spatial structure and boundaries of the tumor, showcasing its ability to differentiate between various tumor types and healthy tissues in a 3D space, achieving an **IoU of 0.6419**.

2D Binary Segmentation



the training and validation accuracy curves show a steady improvement over time, with minimal overfitting, while the loss curves reflect consistent convergence. These results highlight the model's effectiveness in segmenting brain abnormalities with high accuracy.

2D Multiclass Segmentation



Both the accuracy and loss graphs, along with the IoU score, indicate moderate results due to limited training data caused by resource constraints. Despite this, the model effectively identified tumor structures across classes, and with more data and resources, its performance would likely improve significantly.

Conclusion

This project highlights the effectiveness of deep learning models, particularly the U-Net architecture, for automating brain MRI segmentation. The model successfully identified tumor structures across various classes, demonstrating its potential in clinical applications.

Future work should focus on increasing the training dataset and enhancing computational resources to improve model accuracy and robustness, ultimately contributing to better diagnostic precision and treatment planning in neuro-oncology. The full source code for this project is available on GitHub: GitHub Repository.

Contact

hajar.dekdegue.3@gmail.com

https://github.com/AjarHDK