U-NET FOR CARDIAC MRI SEGMENTATION

AKIL Zakaria
ABRIK Samia
EL BOUHALI Adnane
IMA206 Computational Imaging

August 17, 2024

Table of Content

Problem

- Cardiovascular diseases (CVDs) are the leading cause of death globally (31% of annual deaths, WHO).
- Early diagnosis and accurate cardiac function assessment are crucial.
- MRI is vital for visualizing heart structure and function but is labor-intensive and prone to variability.
- Need for automated, precise segmentation of cardiac structures.

Objective

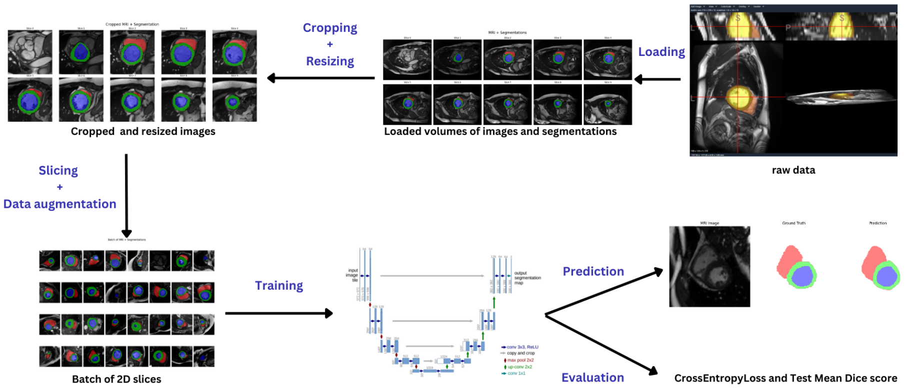
The goal of this project is to develop and evaluate a deep learning model, specifically a U-Net architecture, to automatically segment cardiac structures from MRI images using the Automated Cardiac Diagnosis Challenge (ACDC) dataset. The U-Net model, with its encoder-decoder structure, is well-suited for image segmentation tasks due to its ability to capture fine-grained details and contextual information.

Key Steps

This project aims to achieve the following objectives:

- **Data Preprocessing:** Implement preprocessing steps to prepare the ACDC dataset for training and validation, including normalization, cropping, and slicing of MRI images.
- **Data Augmentation:** Apply various data augmentation techniques to enhance the model's robustness and generalizability.
- **Model Development:** Train a U-Net model to accurately segment key cardiac structures such as the left ventricle, right ventricle, and myocardium.
- **Evaluation:** Assess the model's performance using metrics such as Dice coefficient and Cross Entropy loss.
- **Visualization:** Provide visual representations of the segmentation results to demonstrate the model's effectiveness.

Methodology



Pre-processing

Cropping & Resizing

Objective: Standardize input data size for the U-Net model by focusing on the region of interest (the heart) in MRI images.
Cropping: Focus on the region around the heart.
Method:

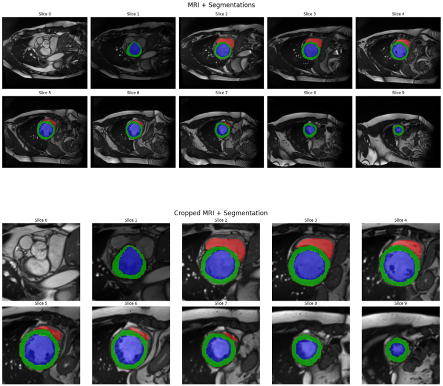
- Find a rectangular bounding box around the heart
- Margin Addition: Added a margin of 20 pixels around the heart to ensure full inclusion.
- Square Crop: Ensured the crop is square to avoid deformation.
- Centering: Calculated the maximum length in the x and y dimensions and centered the crop.

Resizing: Ensures uniformity in input data dimensions while preserving essential features by resizing both images and segmentation masks to 128x128.

Pre-processing

Cropping & Resizing

- **Enhanced focus:** Isolates the heart region, removing irrelevant parts of the image.
- **Square Cropping:** No distortions or deformation
- **Volume Cropping:** Same crop through all the slices of the same volume
- **Uniform Input Size:** Facilitates batch processing and model training by standardizing the input size.



Results of Cropping and Resizing

Transforming Volumes to Slices

- **3D MRI Volumes:** Original cardiac MRI data is in 3D volumes. Each volume consists of multiple slices representing different cross-sections of the heart.
- **Why Slice the Volumes?** Model Compatibility: The U-Net model operates on 2D images. Slicing 3D volumes into 2D slices makes them compatible with the model.

Pre-processing

Data Augmentation

Why data augmentation ?

- Increase Training Set Size
- Improve Model Robustness by introducing Variations
- Prevent Overfitting

Data Augmentation Techniques Used

- Random rotation by 90 degree
- Random flip along the x-axis and y-axis
- Random scaling of the intensity of the image
- Random adjustment of the contrast of the image based on a gamma value

U-Net Architecture

General architecture

In semantic segmentation we find multiple CNN architectures that are efficient and perform well in multiple field especially biomedical imaging. AlexNet, ResNet, VGG-16 and GoogLeNet and for sure the U-Net.

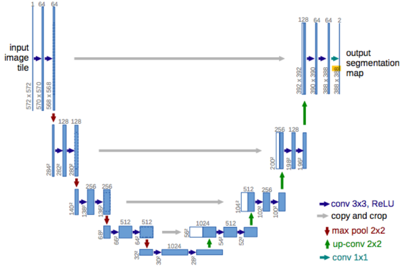


Figure (1): Example of U-net architecture

U-Net Architecture

General architecture

As a variant of UNet architecture, we have 3D Unet that consists on 3dconvolutions rather than 2 dimensional ones.

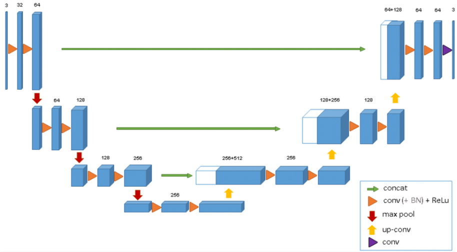


Figure (2): Example of 3dU-net architecture

- **Data Splitting:**
  - **Training Set:** 70%
  - **Validation Set:** 30%
- **Batch Size:** 32
- **Epochs:** 50
- **Optimizer:** Adam (Learning Rate: 0.001)
- **Loss Functions:**
  - Cross-Entropy Loss
  - Dice-Cross-Entropy Loss

This setup ensures efficient training, stable gradient updates, and effective convergence.

- Training Milesial model on preprocessed data with a CE Loss.

Figure (3): Evolution of training/validation loss

Training

Results - Data Augmentation

- Performing Data augmentation technique before training.

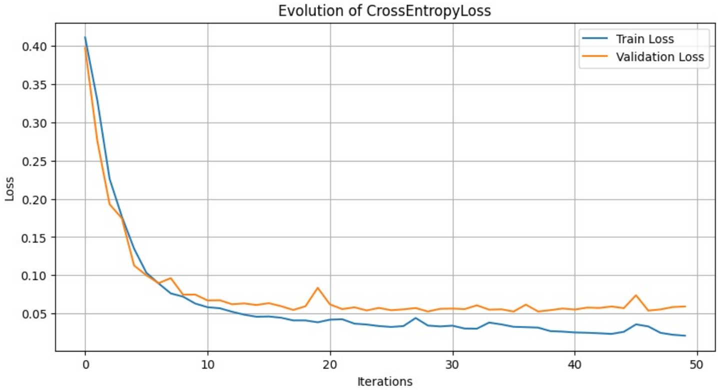


Figure (4): Evolution of training/validation loss

Training

Results - Data Augmentation - DiceCE Loss

- Changing the training loss criteria from Cross-Entropy to DiceCE that combines it with Dice score criteria .

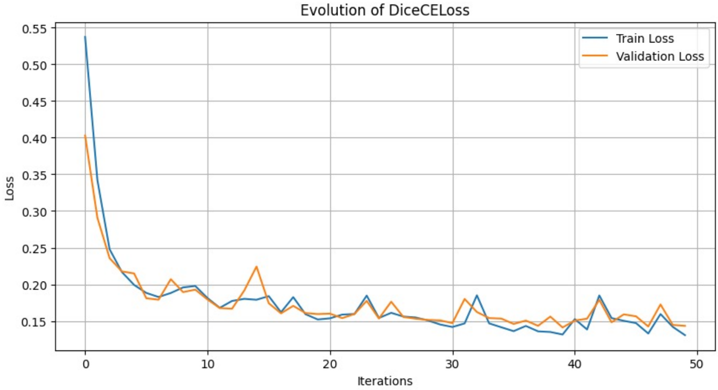


Figure (5): Evolution of training/validation loss

Testing

Results

Model	Data Augmentation	Epochs	Loss Function	Test Loss	Test Mean Dice Score
U-Net	Yes	50	Cross Entropy Loss	0.0871	0.8630
U-Net	No	50	Cross Entropy Loss	0.0715	0.8423
U-Net	Yes	50	Dice CE Loss	0.1597	0.8820

Table (1): Performance of U-Net Models

Testing

Best Result

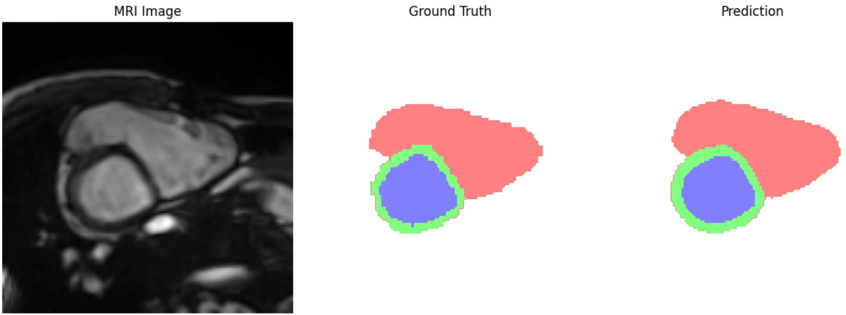


Figure (6): Comparing the result of our segmentation with the ground truth

```
100%|██████████| 34/34 [00:05<00:00,  6.41it/s]
Test Mean Dice: 0.8820
```

Figure (7): Best Dice Score

- A more interesting testing performance would be to test without using test ground truth to transform test input (cf. our best result).

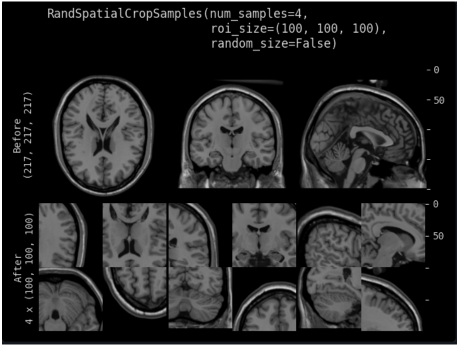
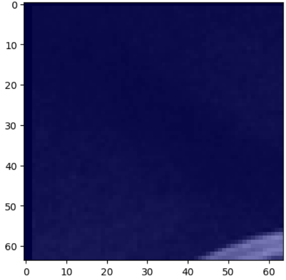


Figure (8): RandSpatialCropSamples from monai

- We perform a patch sampling with the same size for all the patchs.



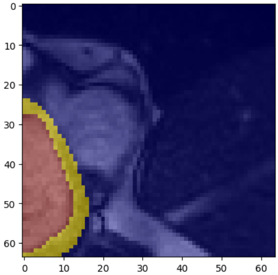
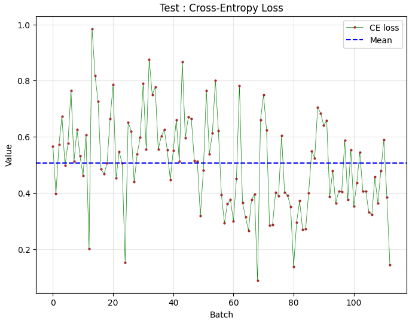


Figure (9): Examples of two patchs from a slice

Patch Sampling - Results

- We perform a patch sampling with the same size for all the patchs.



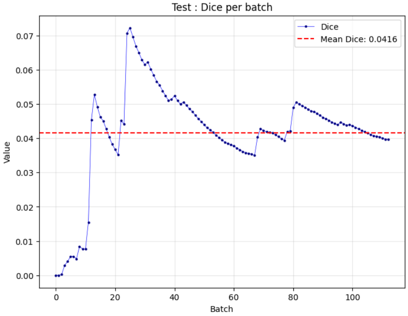


Figure (10): Evolution over test batches

Conclusion

Good performance and suggested improvements

The segmentation accuracy of the given method is around 88% achieved.

These limitations prompt future efforts to focus on:

- **Hyperparameter Tuning:** Experiment with different learning rates, batch sizes, and optimizer configurations.
- **Model Architecture Enhancements:** Integrate skip connections, attention mechanisms, or use more advanced architectures like U-Net++ or Attention U-Net.
- **Ensemble Methods:** Combine predictions from multiple models to improve robustness and accuracy.
- **Practical Testing Constraints:** In practice, we lack ground truth segmentations for the test set, rendering the cropping technique unfeasible. Instead, we should explore evaluating the model's performance using patches sampled directly from the target image (patch sampler).

References

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. CoRR, abs/1505.04597, 2015.
[2] Pytorch-UNet GitHub Repository