Pusan National University

Computer Science and Engineering

Technical Report 2024-10

Improving the Generalizability of Multi-Center and Multi-Vendor Cardiac MRI Segmentation Models

지도교수: 감진규

202055633 이슬람 살리흐

202155631 케네스 예라슬

202155629 누가예바 알트나이

Table of Contents

Improving the Generalizability of Multi-Center and Multi-Vendor Cardiac MRI Segmentation Models 1

1. Introduction 4

1.1 Introduction and Problem Background 4

1.2 Objectives 5

2. Literature Review 6

2.1 Generalizability in Medical Imaging 6

2.2 Segmentation Models for Cardiac MRI 6

3. Dataset 7

3.1 Dataset Exploration and Analysis 7

3.2 Dataset Characteristics 8

3.3 Data Preprocessing 9

4. Methodology 10

4.1 Baseline Model 10

4.2 Semi-supervised Learning with QC Candidate Selection 12

4.2.1 Supervised Learning 12

4.2.2 Convolutional Autoencoder 13

4.2.3 Semi-supervised Learning 14

4.3 U-net based Domain Adversarial Neural Network (DANN) 16

4.3.1 DANN Architecture 18

5. Experiments and Results 19

5.1 Evaluation Metrics 19

5.2 Results for Semi-Supervised learning with QC Candidate Selection 21

5.3 Results for U-net based DANN 23

5.4 Comparison of Models 25

6. Production Deployment 27

7. Updated Project Implementation Plan 28

7.1 Updated Development Schedule 28

7.2 Role Division 35

8. Conclusion 39

8.1 Results Discussion 39

8.2 Future Work 40

References 40

1. Introduction

1.1 Introduction and Problem Background

Cardiac magnetic resonance (cMRI) is known as the most acknowledged standard for the clinical evaluation and assessment of various cardiac diseases. When observing cMRIs, it is very important to gather information about the left ventricle (LV), myocardium, and right ventricle (RV). This is needed to identify the type of disease and its precise location. Although traditional manual segmentation methods are still widely used, it is very likely that the results of this assessment will greatly depend on the experience of each professional and the conditions in the hospital. Moreover, manual segmentation requires careful examination of each scan and often needs a lot of time, which probably will lead to errors and inconsistencies. The process requires extensive manual input to annotate cardiac boundaries across all image slices and phases, potentially compromising the consistency and accuracy of the result (Campello et al., 2021). This further proves the idea of improving and integrating automated or semi-automated assessment techniques to enhance the efficiency and reliability of cardiac assessment in clinical practice.

As a solution to the limitations of manual segmentation, deep learning-based methods are becoming increasingly popular as they provide more accurate and effective solutions to the problem. For example, the usage of CNN by Tran et al. shows a significant advantage in the segmentation of the left and right ventricles in short-axis (SA) MRI scans (Tran, 2017). Following this trend, Poudel introduced a recurrent FCN network that works with spatial information to improve the performance of left ventricle (LV) segmentation (Poudel et al., 2017). Although this method is developing very fast, we are still lacking precision and accuracy in the segmentation of the right ventricle (RV) as it is challenging because of its heterogeneous intensity, variable morphology, and indistinct boundaries in MRI scans. These challenges are further complicated in the basal and apical slices, where existing methods frequently suffer from reduced segmentation accuracy (Li et al., 2022).

Nonetheless, the main limitation of current deep learning models is their generalization. It often struggles when facing previously unseen datasets from different clinical centers or imaging vendors. As this kind of variability in imaging protocols and patient demographics or disease characteristics, and scanner-specific biases will occur constantly in real-life clinical practices, it can significantly degrade the overall performance and reliability of these models (Glocker et al., 2019).

Automating cardiac segmentation is very important for the efficient and reliable diagnosis of cardiovascular diseases. Even though deep learning methods have shown big potential, their limitations require further research

in this field. There is an urgent need to develop models with enhanced robustness and generalization capabilities, which will consistently handle the variability of multi-center and multi-vendor datasets.

In this paper, we investigated different strategies in improving the generalizability of Cardiac MRI segmentation including semi-supervised learning, reconstruction training with auto encoder and domain adversarial neural network. In addition, we evaluated and compared the capabilities of each method on different tasks.

1.2 Objectives

The primary objectives for our project are as follows:

1. To conduct an extensive literature review on the cardiac MRI segmentation methodologies, by assessing their strength and limitations. We will particularly assess their generalization across diverse datasets. This objective aims to establish a comprehensive understanding of the problems that need to be solved in this domain.

2. To perform efficient and accurate analysis of the multi-center and multi-vendor datasets, we must implement different data preprocessing techniques such as normalization, and standardization to prepare the dataset for subsequent model training.

3. To visualize the cardiac MRI images and given segmentation labels to gain deeper insights into the dataset's characteristics, thereby preparing for model development and evaluation steps.

4. To develop a new and reliable deep learning-based segmentation model with the objective of improving generalization across multi-center and multi-vendor datasets, addressing the problem of constant variability of these datasets.

5. To evaluate and compare the performance of different models on a chosen dataset for its ability to generalize across different clinical centers and vendors. This evaluation will include a deeper analysis of initial results and further improvements, mainly focusing on the accuracy and reliability of the model.

6. To solve the generalizability issues of deep learning models on the segmentation process, different methods like domain adaptation and data augmentation will be used. To assess the effectiveness of these methods, we will conduct comprehensive testing and analysis.

7. To provide practical usage of created models in clinical practices aimed at increasing the efficiency and reliability of deep learning models in diagnostic image workflow in cardiac imaging.

2. Literature Review

2.1 Generalizability in Medical Imaging

Researchers working on deep learning models mostly train and evaluate their deep learning models on their own datasets, but this approach doesn't provide insights into the model's ability to generalize to data from other institutions, which may exhibit different underlying distributions. A more robust practice involves testing the model on an external dataset, a requirement that some journals have recently implemented for published deep learning research (David et al., 2020). While using one external dataset is better than none, it still falls short of fully demonstrating the model's generalizability.

Consider, for instance, a recent study on deep learning for breast cancer screening (McKinney et al., 2020). The researchers trained their model on two datasets from the UK and tested it on a single dataset from the US, subsequently claiming that their model could generalize between these countries. However, the US dataset was sourced from just one institution, with 99% of the data collected using scanners from the same vendor. This limited evaluation—based on a single vendor and institution—fails to represent the vast range of clinical environments, data distributions, scanner types, and imaging protocols that exist across institutions.

To address the challenge of generalizability, many researchers aim to gather as much diverse patient data as possible, hoping to train a deep learning model that performs reliably in any clinical setting, for any patient, at any time. However, this goal is highly ambitious, and practically unattainable, given the difficulty of collecting sufficient data from a broad range of medical institutions to account for all possible clinical scenarios.

2.2 Segmentation Models for Cardiac MRI

There are many architectures and techniques developed by researchers to address the problem of Cardiac MRI segmentation. Unlike usual image segmentation, medical image segmentation requires the ability to work with 3D (image slices of human organ) or even 4D (image slices across some period) data. As Paschali, M. et al. stated heavy data objects result in overparametrized deep learning models which *"memorizes"* the training data.

The early mentioned Fully Convolutional Neural Networks presented by Poudel et al. and Tran shows the state-of-art results on Left Ventricular and Right Ventricular. However, their model heavily relies on the diverse and relevant data fed to the model. In medical imaging, acquiring reliably annotated and diverse dataset is very difficult task. Sharing medical data is significantly more complex and challenging compared to sharing real-

world images. Data privacy concerns encompass both sociological and technical dimensions, and addressing these issues requires a coordinated effort from both perspectives (Razzak et al. 2018)

Along with trivial Deep Convolutional Networks and U-net models (which is popular in medical image segmentation) some researchers used more advanced techniques. Beetz, M. et al, for example, utilized Crossover Attention over U-net Cascade with Fourier Domain Adaptation. Such complex framework allowed them to effectively carry out semantic segmentation over dilated right ventricle which thought to be a challenging task. Their model showed improved results over trivial U-net based models, especially in segmenting disease distorted images in MnM's 2 dataset. Nevertheless, experiments showed that while U-net cascade performs well on Long Axis, the results on Short Axis presented increased Hausdorff Distance. Beetz, M. et al. suggests that performance discrepancy may be due to the use of 3D-based metrics for the SA stack, which capture errors across all three spatial dimensions, unlike the 2D-based metrics used for long-axis data.

3. Dataset

3.1 Dataset Exploration and Analysis

We have chosen the M&Ms-2 Challenge dataset for our project, which contains the results of heart MRI of 360 patients. The number of patients includes individuals who suffer diseases that affect the right ventricle and left ventricle. There are also healthy individuals, which will help the model to be more generalized. This dataset is an ideal match for model creation due to its multi-center and multi-vendor nature. All the subjects were scanned at 3 clinical centers in Spain using MRI scanners from three different vendors: Siemens, General Electric, and Philips Medical Systems (Martín-Isla et al., 2023).

The MRI data were acquired using 9 different scanner models across the three vendors:

1. **General Electric (GE Healthcare)**: SIGNA EXCITE, Signa HDxt, Signa Explorer.

2. **Philips**: Achieva.

3. **Siemens**: SymphonyTim, Symphony, TrioTim, Avanto Fit Avanto.

These scanners vary in their imaging capabilities, technologies, and specific features used, which will create the range of imaging conditions that may be encountered in clinical practice. This is important to us as it creates

multi-vendor variability, which is a significant factor in developing robust segmentation models that can generalize well across different image settings.

In total, the dataset includes 160 training and 40 validation instances covering 8 types of pathologies. The training set consists of 200 annotated images from four distinct centers. Experienced clinicians segmented the cardiac MRI (CMR) images, delineating the contours for the left ventricle (LV) and right ventricle (RV) blood pools, as well as the left ventricular myocardium (MYO). The labels used in the annotations are as follows: 1 for LV, 2 for MYO, and 3 for RV. These annotations are provided in both short-axis and long-axis views, encompassing a range of challenging RV pathologies and LV remodeling.

3.2 Dataset Characteristics

Annotation Process

Each CMR study was manually annotated by expert clinicians with experience ranging from 3 to over 10 years. Annotations were performed following clinical protocols, focusing on the end-diastolic (ED) and end-systolic (ES) phases in short-axis views. The annotated regions include the left and right ventricles (LV and RV) cavities and the left ventricle myocardium (MYO).

Standard Operating Procedure (SOP)

To ensure consistency and accuracy in annotations, the Standard Operating Procedure (SOP) from the ACDC challenge was adopted. The SOP provides a standardized set of instructions for executing specific tasks, ensuring that all annotations meet high standards of quality and consistency.

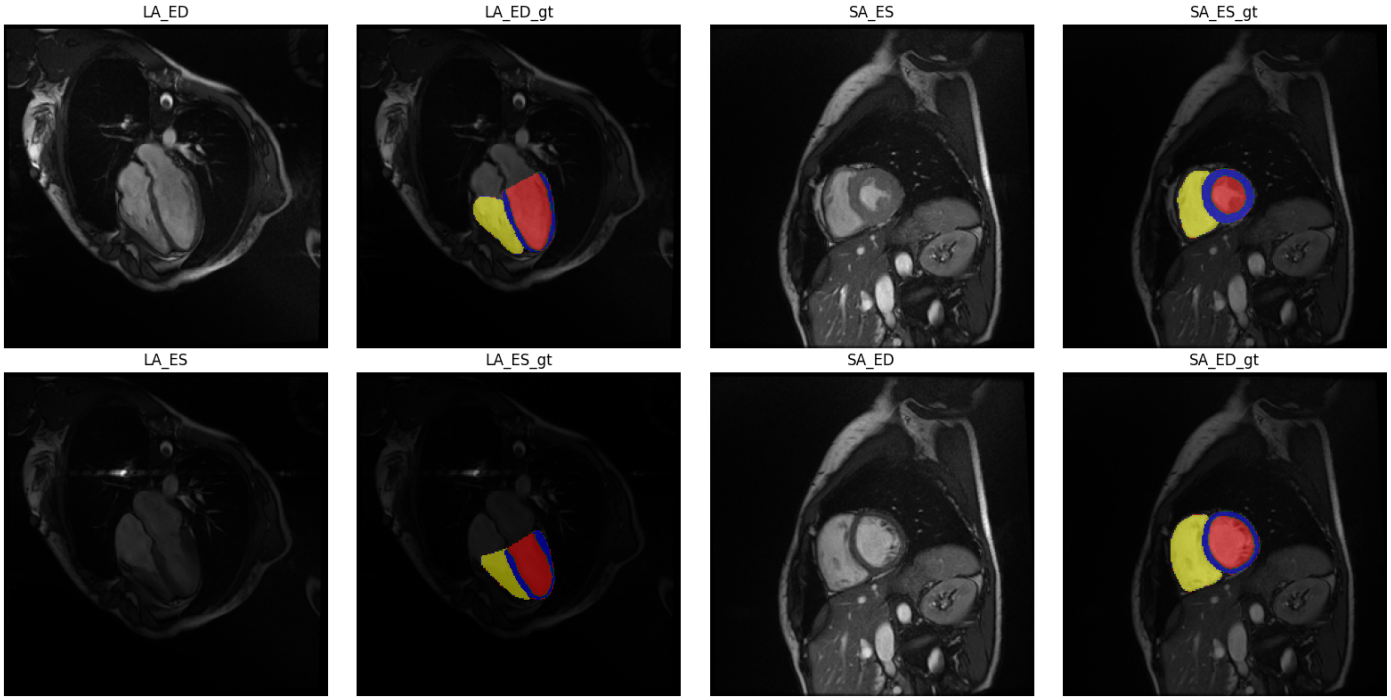


Figure 1. Sample images from dataset representing LA and SA in End-Diastole and End-Systole. Red, Yellow and Blue areas in Ground Truth shows LV, RV, and MYO respectively

3.3 Data Preprocessing

The preprocessing of cardiac MRI images is essential for preparing the data for deep learning applications, especially for segmentation tasks. In this study, we implemented a structured approach to enhance data quality and standardization.

Initially, each image is cropped to focus on the region of interest, typically covering around 80% of the original volume, based on patient-specific information. This is followed by transposing the images to align with the expected input shape of deep learning models.

The voxel spacing of the original images is then updated to match a predefined target spacing, crucial for ensuring consistency. For instance, images with a voxel size of 1.0 mm are resized to a target of 0.5 mm, enabling a standardized resolution that enhances feature learning.

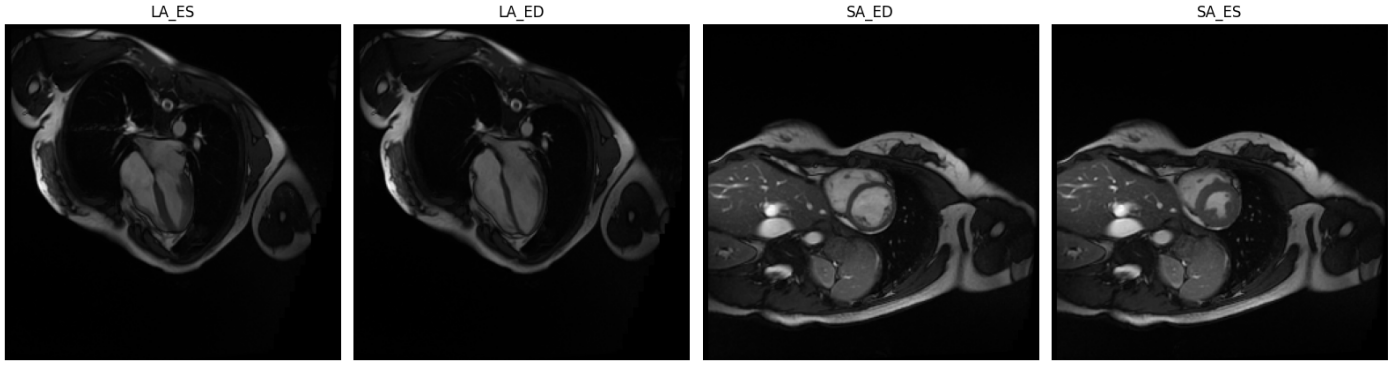


Figure 2. MRI slices before pre-processing

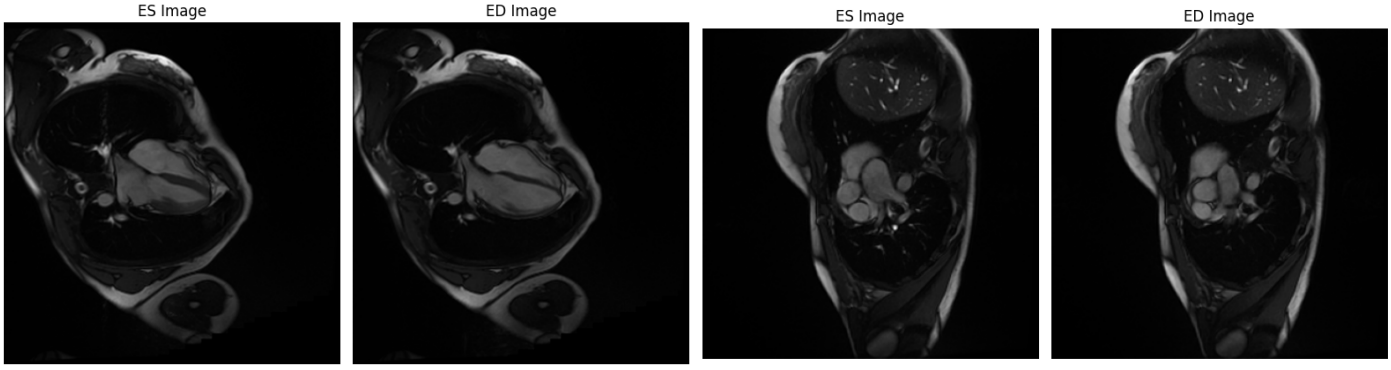


Figure 3. MRI slices after pre-processing

4. Methodology

In this paper, two models for training are investigated. First one is Semi-supervised Learning with Quality Control Candidate Selection suggested by Galati, F. et al, and second one is U-net based Domain Adversarial Neural Network. The details of the training process will be discussed later in this section.

4.1 Baseline Model

For baseline model for both techniques in this paper we utilized Generic U-net model. Since, U-net is developed primarily for medical image segmentation tasks (Ronneberger et al. 2015), its value and efficiency in this study would be crucial.

The U-Net architecture is structured around two main paths. The first is the contracting path, often called the encoder or analysis path, which operates similarly to a conventional convolutional network, focusing on extracting classification features. The second path is the expanding path, also referred to as the decoder or synthesis path, which incorporates up-convolutions and merges features from the contracting path. This

expansion not only restores the spatial resolution of the output but also enables the network to learn detailed, localized features. The final layer in this path typically performs a convolution to produce the fully segmented image. The architecture's nearly symmetrical design gives it a U-like appearance. Unlike traditional convolutional networks, which are primarily designed to classify entire images into single labels, U-Net is specifically built to capture pixel-level context, a crucial aspect for tasks like medical image analysis.

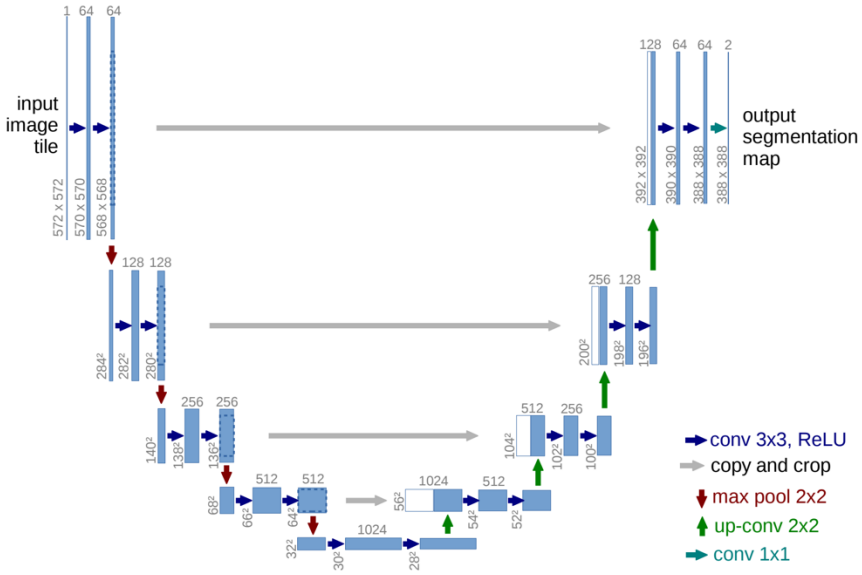


Figure 4. U-net architecture (example for 32x32 pixels in the lowest resolution). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations (Ronneberger et al. 2015).

What makes U-Net particularly valuable is its ability to generate highly detailed segmentation maps even when using a limited amount of training data. This feature is especially important in the medical imaging field, where acquiring properly labeled datasets is often challenging. U-Net addresses this issue by applying random elastic deformations to the training data, allowing the network to learn variations without the need for additional labeled images. This technique enhances the model's ability to generalize from small datasets, making it a powerful tool for medical image analysis.

For this model, Stochastic Gradient Descent was used as optimizer, with weight decay $3 * 10^{-5}$ and momentum 0.99.

As mentioned in the objectives of this research and as its topic states, we aim to improve the generalizability of our models. To test this capability, we excluded Philips vendor from all training and validation data, leaving it to testing part. This way, we will be able to challenge our models against previously unseen data and test the generalizability.

4.2 Semi-supervised Learning with QC Candidate Selection

In this section, we explore a semi-supervised learning approach incorporating Quality Control (QC) Candidate Selection. Semi-supervised learning is particularly useful when labeled data is limited, as it leverages both labeled and unlabeled data to improve model performance (Galati et al. 2022). By introducing QC Candidate Selection, we ensure that the most informative and high-quality samples are chosen for training, enhancing the model's learning efficiency and generalizability. In the implementation by Galati et al. the core focus was directed on segmentation of RV, however in our modified model we tried to adapt it to perform well on RV, LV and MYO.

The learning process consists of three integral parts: Supervised training, Convolutional Autoencoder and Semi-supervised training.

4.2.1 Supervised Learning

During the supervised learning data is trained on Generic U-net model. In this phase all training data is labelled, and the model is trained to minimize the following loss function:

$$L_{sup} = L_{GD}(y_{pred}, y_{gt}) + L_{CE}(y_{pred}, y_{gt})$$

Here L_{sup} is the dissimilarity between predicted segmentation and ground truth segmentation. It is equal to the sum of (L_{GD}) Generalized Dice Loss and (L_{CE}) Cross-Entropy Loss. Overall structure can be observed in Figure 5.

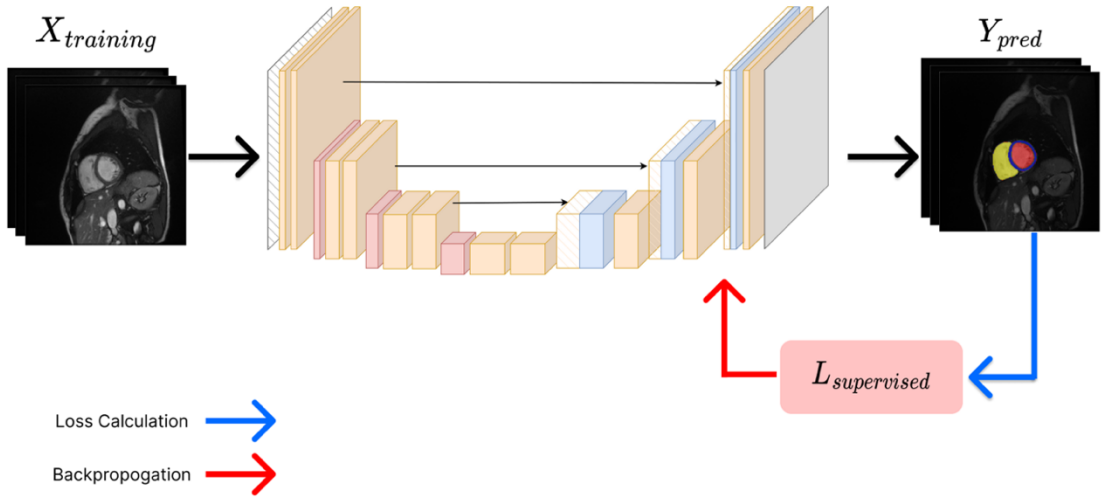


Figure 5. Visualization of Supervised learning phase

4.2.2 Convolutional Autoencoder

For reconstructing purposes, we adjusted the Convolutional Autoencoder (CA) model suggested by Galati, F., Zuluaga, M.A. It is trained on in-distribution[1] samples from the training dataset to reconstruct pseudo ground truth.

Layer	Output Size	Parameters		
		Kernel	Stride	Padding
Input	256x256x4			
Block1	128x128x32	4x4	2	1
Block2	64x64x32	4x4	2	1
Block3	32x32x32	4x4	2	1
Block4	32x32x32	3x3	1	1
Block5	16x16x64	4x4	2	1
Block6	16x16x64	3x3	1	1
Block7	8x8x128	4x4	2	1
Block8	8x8x64	3x3	1	1
Block9	8x8x32	3x3	1	1
Conv2d	4x4x100	4x4	2	1

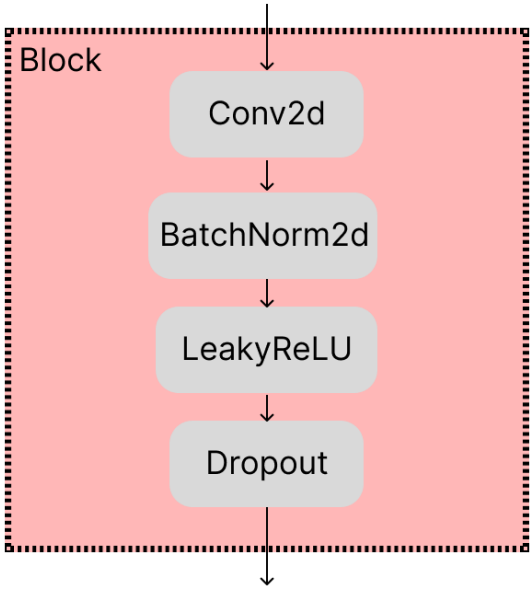


Figure 6. Architecture design of Encoder part

In Figure 6, the encoder architecture of CA is presented. The decoder part of CA is built by mirroring the encoder design and replacing convolution with transposed convolutions. The backbone for CA is taken from Bergmann et al. The core idea behind this approach is to estimate a model that captures the variability in cardiac segmentation masks using a reliable reference training dataset with accurate ground truth labels.

[1] **In-distribution data** refers to data that comes from the same distribution as the training data

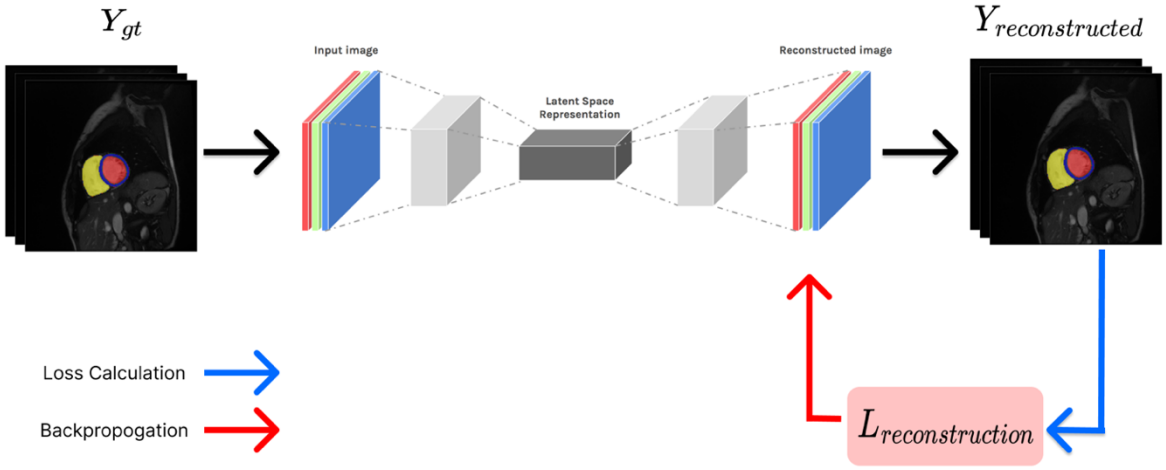


Figure 7. Visualization of Reconstruction phase

As shown in the Figure 7, model takes ground truth labels from the training dataset and learns to reconstruct reliable in-distribution labels. The loss function used is following:

$$L_{rec} = L_{MSE}(y_{rec}, y_{gt}) + L_{GD}(y_{rec}, y_{gt})$$

Where L_{MSE} is Mean Squared Error loss and L_{GD} is Generalized Dice loss. By taking similarity degree between y_{rec} and y_{gt} we acquire the surrogate measure for Quality Control of the segmentation, which also serves as an indicator of the model's performance on unseen data. When the similarity is low ($y_{rec} \approx y_{pred}$), the segmentation is deemed inadequate, and the sample is flagged as a potential out-of-distribution (OoD) case. For these flagged samples, the QC feedback is used to backpropagate and improve the model. The full procedure is explained in the next section.

4.2.3 Semi-supervised Learning

Semi-supervised Learning is accomplished by incorporating Supervised Learning and Reconstruction model together.

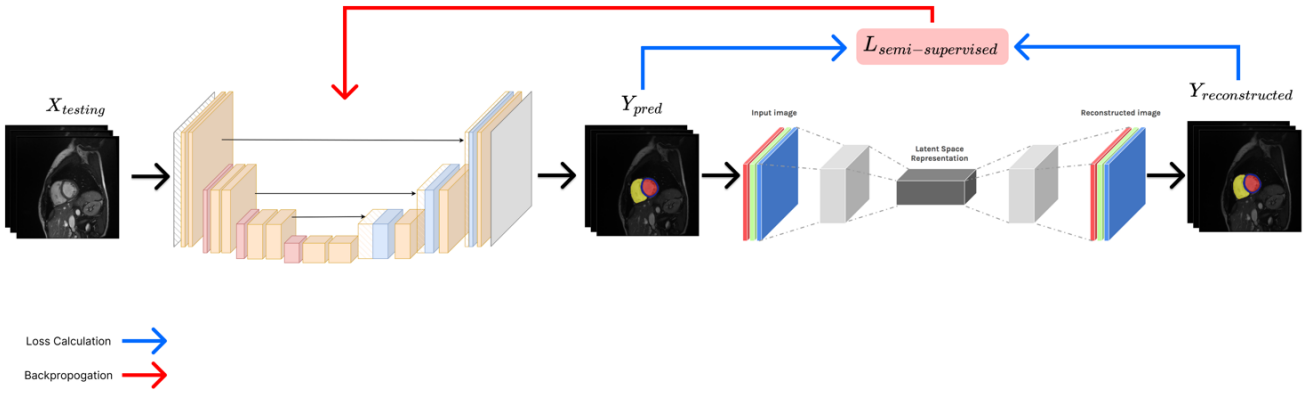


Figure 8. Visualization of Semi-supervised Learning phase

When Supervised Model segments the image, the results will be fed to Reconstructor model. As we discussed earlier, the primary objective of Reconstructor model is to identify OoD data. If the OoD data has been found, the feedback on loss function is backpropagated both from Y_{pred} and $Y_{reconstructed}$ (see in Figure 8). Here, the loss function has the following definition:

$$L_{semi} = \alpha L_{WGD}(y_{rec}, y_{pred}) + L_{sup}$$

In this equation, L_{WGD} is weighted generalize Dice loss, and L_{sup} is loss function for supervised training that we defined earlier. The parameter α plays significant role, accounting for controlling the reliability of Reconstruction model. In learning process, as more OoD data is met, the feedback from Reconstruction model becomes less helpful and in could potentially reinforce errors. α is set to $\frac{1}{k}$, where k is number of epochs. This setting allows better feedback from reconstructor at the beginning (when it is highly reliable) and downgrades its value to the end (when it is not reliable).

The problem of identifying OoD data is rather tricky, and Convolutional Autoencoder alone might not be sufficient. In this paper, we adapted method proposed by Galati et al. The idea is to measure Dice Coefficient and Hausdorff Distance between reconstructed and predicted segmentation, let us label them pseudo Dice Coefficient (pDC) and pseudo Haudorff Distance (pHD). For given set of estimated pDC and pHD we compute the first and third quartiles, which we will us to set up the following lower and upper thresholds:

$$th_{low} = Q_1^{pDC} - 1.5(Q_3^{pDC} - Q_1^{pDC})$$

$$th_{high} = Q_1^{pHD} - 1.5(Q_3^{pHD} - Q_1^{pHD})$$

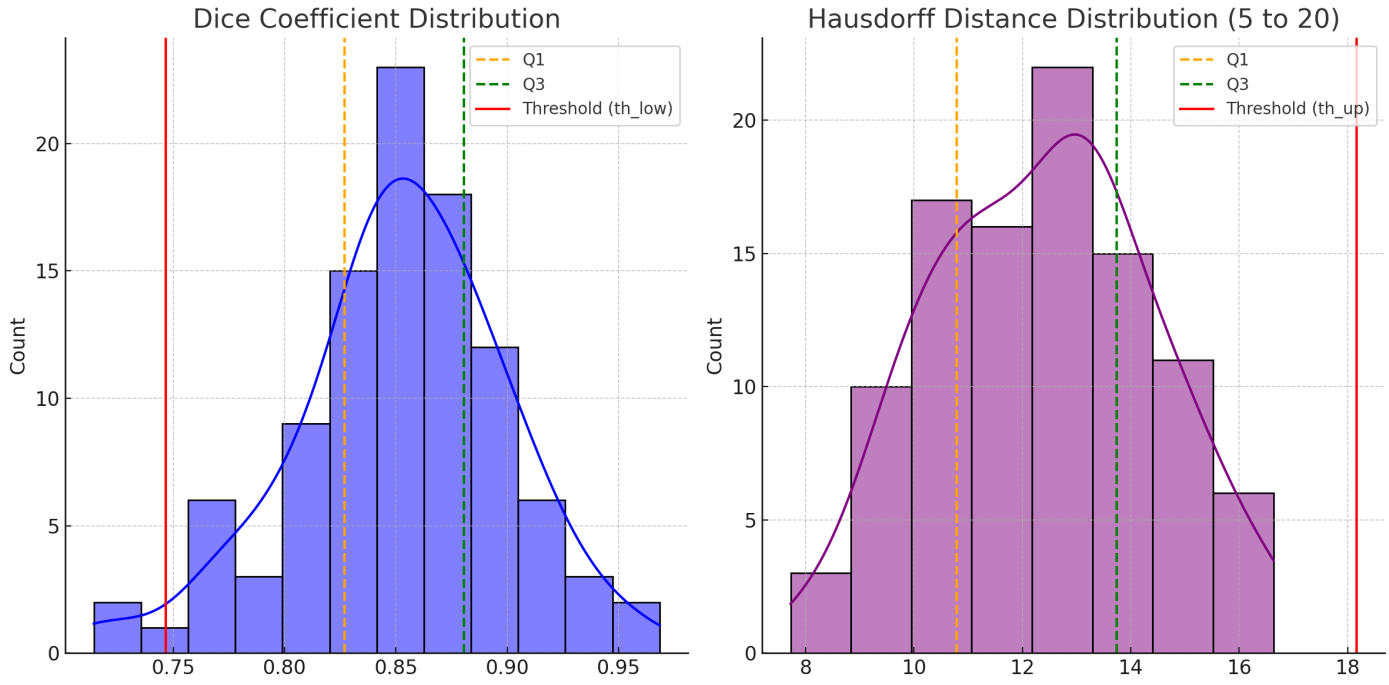


Figure 9. pDC and pHD distributions with Upper and Lower thresholds (red line).

As shown in Figure 9, if the segmentation has pDC lower than th_{low} or pHD higher than th_{up}, then it is identified as OoD data.

It is worth to mention that we save the model at certain times during training. After semi-supervised refinement, the best segmentation prediction is selected through a Quality Control (QC)-based candidate selection process. This involves ranking the predicted segmentations from the best-performing candidate models based on their Dice Coefficient (DC) scores. The top-ranked prediction is then checked against the Hausdorff Distance (HD) threshold. If the HD value is acceptable, this prediction is selected as the result. If not, the next best model is considered, and this process is repeated. If none of the models meet the required thresholds, the final segmentation is set as the average of the best predictions according to both DC and HD.

4.3 U-net based Domain Adversarial Neural Network (DANN)

One of the key approaches to address the problem of model generalization is domain adaptation. In machine learning, it allows models to generalize across different datasets when there is a shift in data distribution between the source and target domains. In cardiac MRI segmentation, domain adaptation is particularly crucial due to the variability introduced by different vendors, imaging centers, and clinics. Each MRI scanner, depending on its manufacturer or settings, may produce images with differing characteristics such as resolution, contrast, and

noise levels (H. Guan and M. Liu, 2022). These variations can severely impact the performance of models trained on a specific dataset when applied to new, unseen data from other sources.

In this paper we effectively used the Domain Adversarial Neural Network. It combines feature learning with adversarial training, where the network is trained to perform two tasks: one, to predict the main target (image segmentation in our case), and two, to confuse a domain classifier that tries to distinguish between data from different domains. By doing this, DANN forces the model to learn domain-invariant features, enabling it to work well across various domains. MnM's 2 dataset provides clinical images from different vendors, which we considered as domains in this research.

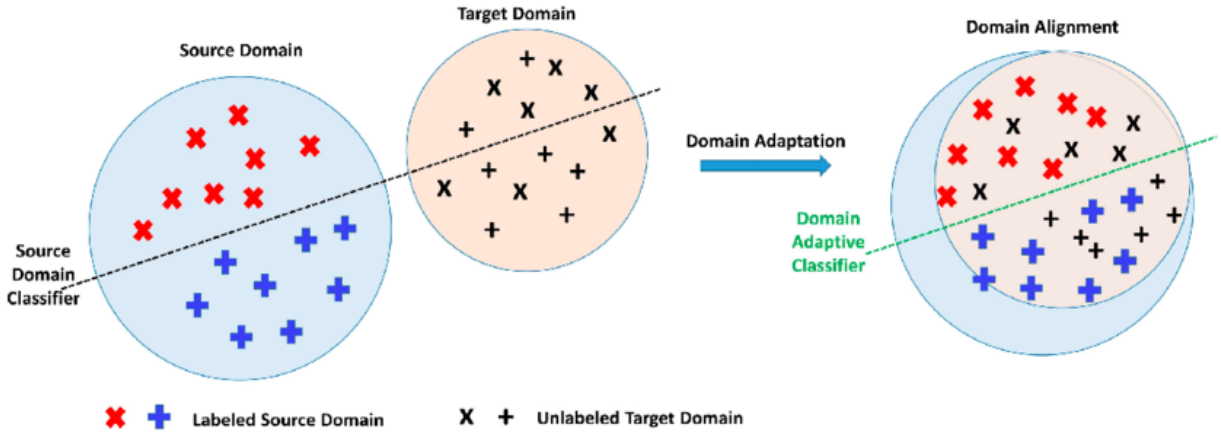


Figure 10. Example of allignment of features after Domain Adaptation (Goel et al. 2023).

4.3.1 DANN Architecture

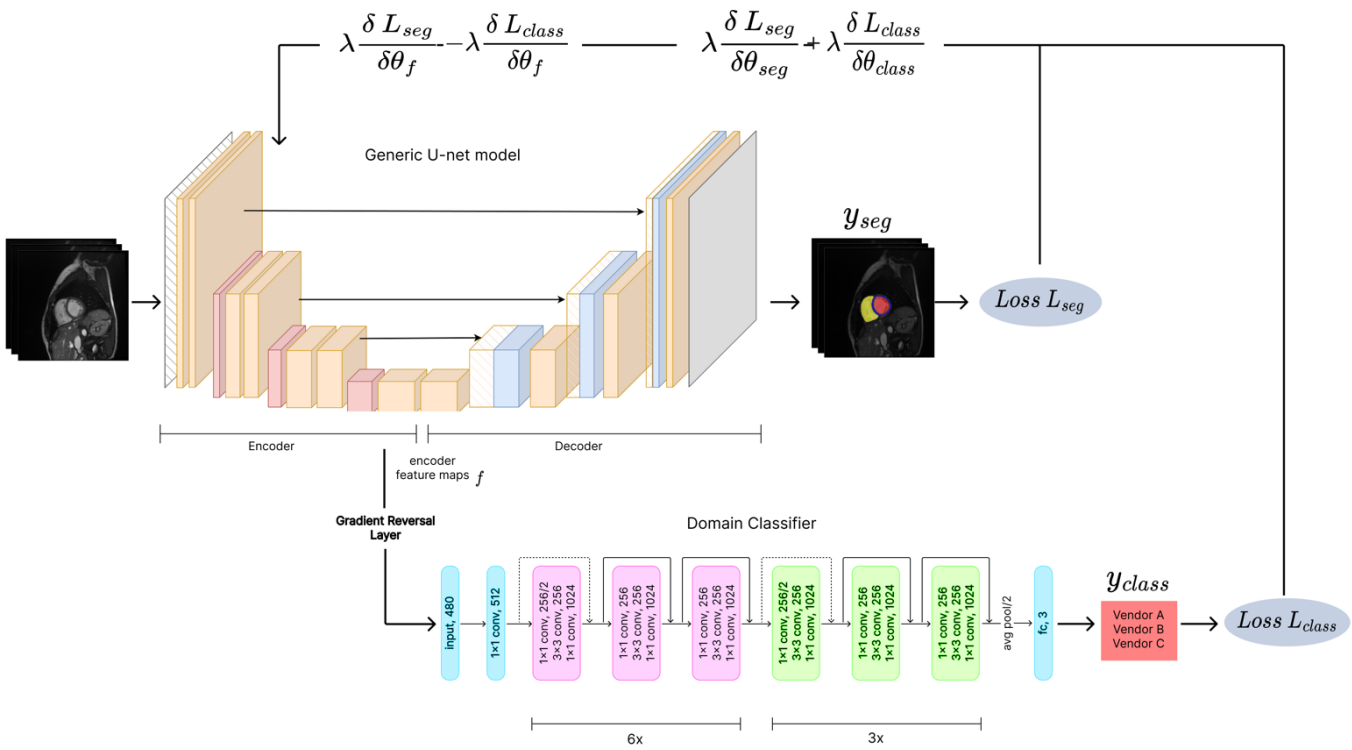


Figure 11. Design of U-net based DANN model

Usually, DANN consists of three main parts. Those are feature extractor, main predictor/segmentor, and domain classificator. However, in our case encoder and decoder of U-net model works as feature extractor and segmentor respectively. For the Domain Classifier, we exploited modified ResNet50 (Zhang et al. 2016) model. The modification in model involved removing all initial layers which were responsible for raw image processing and feature extraction, since it was done by encoder part of U-net. In ResNet50, pivotal layers were those responsible for working with high level abstraction (layers 3, 4) which are shown in Figure 11. In addition, ResNet introduced deep layer convolutional layers for better feature processing without computational load due to residual mapping (Zhang et al. 2016).

Initially, when U-net model was feed forwarded it used to directly pass feature maps from encoder to decoder. In our architecture, it stores the copy of encoder feature maps to further feed forward it to Domain Classifier. Encoder feature map has the output of 7 maps. We are interested in only deeper layers as they store more abstract and domain specific information; therefore, last 3 layers with shapes [20, 480, 16, 16], [20, 480, 8, 8], [20, 480, 4, 4] (Batch Size x Channels x Width x Height) was considered in training. To combine those 3 tensors, we utilized various methods including adding by reshaping and global pooling; however, most prominent method

was feed forwarding them separately and adding their loss values with weights. Weights are given in ascending order starting from shallower layers to deeper to highlight more abstract features. Before passing to Domain Classifier itself, feature map processed through crucial step in DANN – Gradient Reversal Layer.

A key innovation in DANN is the Gradient Reversal Layer (GRL), placed between the feature extractor and the domain classifier. During backpropagation, the GRL reverses the gradient by multiplying it by -λ, encouraging the feature extractor to learn domain-invariant features. This adversarial setup drives the feature extractor to produce representations that are useful for the main task while making the source and target domains indistinguishable (Ganin et al, 2016). Essentially, the domain classifier tries to maximize the difference between domains (H-divergence), while the feature extractor minimizes it, promoting shared feature space across domains.

Choosing the loss function was important task, since the whole idea of creating domain invariant features requires backpropagation from combined loss of both Segmenting module and Domain Classifier.

$$L_{dann} = L_{GD}(x_{seg}, y_{seg}) - \lambda L_{CE}(x_{dom}, y_{dom})$$

It was decided that using Generalized Dice Loss for segmentation (like in in Section 4.2) will be practical, and for domain classification we used Cross Entropy Loss. As it can be observed in equation above, results from two module losses are combined and will be further backpropagated.

5. Experiments and Results

5.1 Evaluation Metrics

1. Dice Score, also known as the Dice Coefficient, is a statistical measure used to evaluate the similarity between two sets, particularly in image segmentation tasks. It ranges from 0 to 1, where a score of 1 indicates perfect overlap between the predicted segmentation and the ground truth, while a score of 0 indicates no overlap. The Dice Score is calculated using the formula:

$$Dice\ Score = \frac{2 \times |A \cap B|}{|A| + |B|}$$

where A represents the predicted segmentation, and B represents the ground truth. This metric is particularly useful in medical imaging, as it provides a balanced measure of accuracy, emphasizing both false positives and false negatives, thereby offering a more comprehensive evaluation of segmentation performance.

2. Hausdorff Distance is a metric used to measure the degree of mismatch between two sets of points, often employed in the evaluation of image segmentation results. Specifically, it quantifies the greatest distance from a point in one set to the nearest point in the other set. In the context of segmentation, it assesses how far the predicted boundaries deviate from the ground truth boundaries.

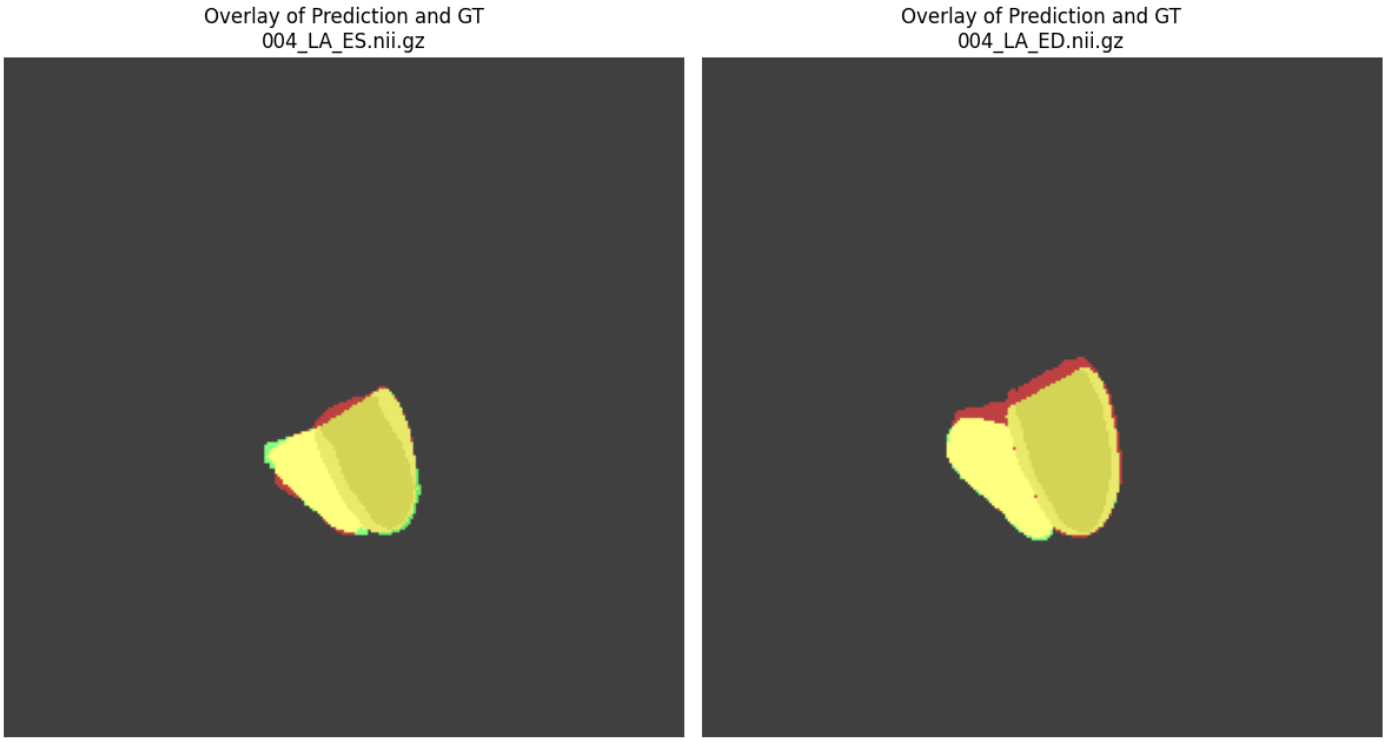


Figure 12. Predicted image overlayed by GT on Long Axis, both ES and ED.

In Figure 12, a predicted label overlayed by ground truth label can be observed. Hausdorff Distance is calculated between points on protruding regions colored in red and green. In this case, the red region on End-Diastole slice will show the highest score.

The Hausdorff Distance is defined mathematically as follows:

$$H(A,B) = \max\{h(A,B), h(B,A)\}$$

where

$$h(A,B) = \max_{a \in A} \min_{b \in B} d(a,b)$$

In this formula, A and B are the two sets of points (e.g., predicted and ground truth segmentation boundaries), and d(a, b) represents the distance between points a and b. The Hausdorff Distance ranges from 0 to infinity, with

a lower value indicating a better alignment between the predicted and true boundaries. It is particularly useful in medical imaging as it captures both the worst-case error and the spatial distribution of discrepancies, providing insights into the accuracy and robustness of segmentation methods.

5.2 Results for Semi-Supervised learning with QC Candidate Selection

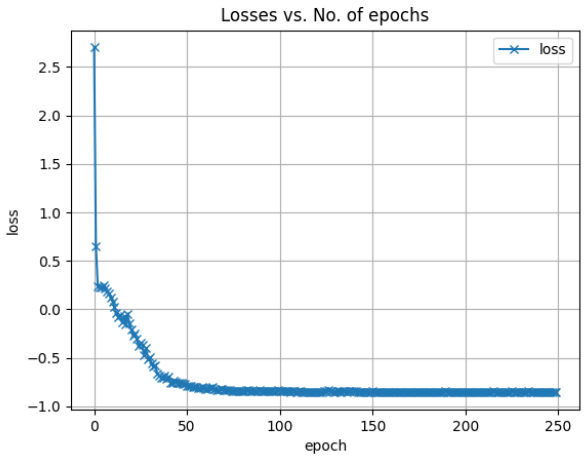


Figure 13. Supervised learning.

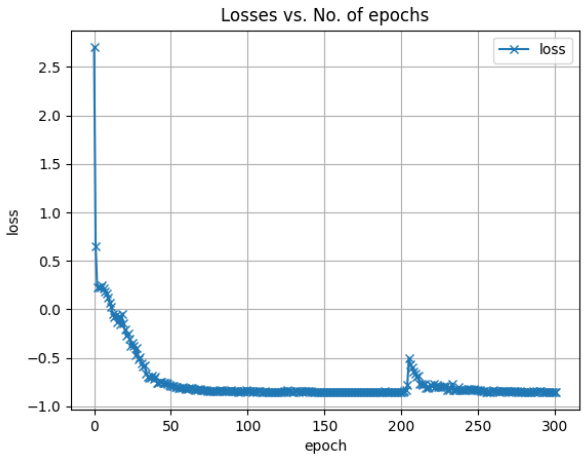


Figure 14. Semi-supervised learning

Both learning phases shows good convergence around 250-300 epochs (Figure 13, 14). In the Figure 14, there is a sharp increase around 200 epochs, which then gently decreased again. It is assumed that Reconstruction model's reliability significantly decreased on that period, and all the OoD data has been investigated. It is worth to consider decreasing the number of epochs on Semi-supervised Refinement to observe the changes.

axis	RV_ED_DC	RV_ED_HD	RV_ES_DC	RV_ES_HD	RV_DC_AVG	RV_HD_AVG
SA supervised	0.7527	14.6867	0.8388	15.0007	0.7958	14.8437
LA supervised	0.8106	15.7189	0.8966	16.0328	0.8536	15.8759
SA semi-supervised	0.8660	8.2020	0.9520	8.516	**0.9090**	**8.3590**
LA semi-supervised	0.8687	5.9658	0.9547	6.2798	**0.9117**	**6.1228**

Figure 15. Dice Score and Hausdorff Distance for RV in ES and ED

The table compares segmentation results for Right Ventricle (RV) on short-axis (SA) and long-axis (LA) slices, using both supervised and semi-supervised learning. The semi-supervised method shows clear improvements on short-axis data, enhancing performance in both RV_ED and RV_ES segmentation. In contrast, supervised results

on LA slices are notably weaker, especially in terms of Hausdorff Distance, which indicates larger variations. However, semi-supervised learning brings improvements on LA data, demonstrating its ability to better generalize in these more challenging cases.

axis	LV_ED_DC	LV_ED_HD	LV_ES_DC	LV_ES_HD	LV_DC_AVG	LV_HD_AVG
SA supervised	0.793	9.831	0.8791	10.145	0.8361	9.9880
LA supervised	0.7648	13.1385	0.8509	13.4525	0.8079	13.2955
SA semi-supervised	0.9004	3.1604	0.9864	3.4744	**0.9434**	**3.3174**
LA semi-supervised	0.9025	4.1929	0.9885	4.5069	**0.9455**	**4.3499**

Figure 16. Dice Score and Hausdorff Distance for LV in ES and ED

In LV metrics our semi-supervised models also performs better, especially in the SA view, where it reduces the Hausdorff Distance significantly (44%) while increasing the Dice Coefficient for both ED and ES phases. The semi-supervised method also enhances results in the LA view but to a lesser extent (3%) compared to the SA view, with small improvements in both Dice scores and Hausdorff Distance.

axis	MYO_ED_DC	MYO_ED_HD	MYO_ES_DC	MYO_ES_HD	MYO_DC	MYO_HD
SA supervised	0.7108	9.53	0.7968	9.844	0.7538	9.6871
LA supervised	0.8067	12.417	0.8927	12.731	0.8497	12.574
SA semi-supervised	0.8144	6.3911	0.9004	6.7052	**0.8574**	**6.5482**
LA semi-supervised	0.9049	4.1384	0.9909	4.4525	**0.9479**	**4.2955**

Figure 17. Dice Score and Hausdorff Distance for MYO in ES and ED

The semi-supervised approach demonstrates clear improvements over the supervised one, particularly in lowering the Hausdorff Distance for both axes (7.6871 $\Rightarrow$ 6.5482, 10.574 $\Rightarrow$ 4.2955). In the SA view, there are slight gains in the Dice Coefficient and a notable reduction in the Hausdorff Distance. The LA view, however, shows more significant improvements (10.6%), with a substantial increase in the Dice score and a marked decrease in Hausdorff Distance, indicating enhanced performance in both the ED and ES phases.

5.3 Results for U-net based DANN

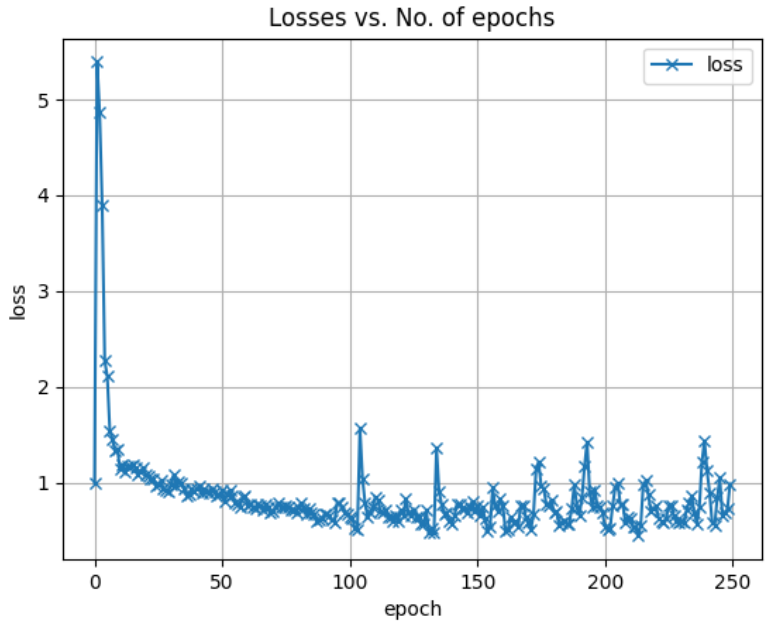


Figure 18. U-net training with DANN

The results for loss value shows overall good tendency, however the model did not fully converged at the end. We speculate that these fluctuations in loss function may be caused by imbalanced dataset. Across 3 vendors, Siemens occupies more than 60% of all dataset, resulting in uneven class distribution for domain classifier.

axis	RV_ED_DC	RV_ED_HD	RV_ES_DC	RV_ES_HD	RV_DC_AVG	RV_HD_AVG
SA	0.8941	7.0345	0.9004	7.1955	0.8945	7.1946
LA	0.8567	17.200	0.8463	17.672	0.8468	17.827

Figure 19. Dice Score and Hausdorff Distance for RV

For the Short Axis (SA) view, the model achieved strong Dice Scores with 0.8941 for End-Diastole (ED) and 0.9004 for End-Systole (ES), indicating a high degree of overlap between the predicted and ground truth masks. The Hausdorff Distances for SA, however, are moderate at 7.03 mm for ED and 7.19 mm for ES, suggesting reasonable boundary matching, though there is still room for improvement in boundary precision.

In contrast, the Long Axis (LA) results show a noticeable drop in performance. Dice Scores are lower, with 0.8567 for ED and 0.8463 for ES, indicating reduced overlap accuracy. The Hausdorff Distances are significantly higher for LA, at 17.20 mm for ED and 17.67 mm for ES, reflecting much poorer boundary alignment in the LA view.

axis	LV_ED_DC	LV_ED_HD	LV_ES_DC	LV_ES_HD	LV_DC_AVG	LV_HD_AVG
SA	0.9218	4.9304	0.9398	5.8901	0.9300	5.0078
LA	0.9238	9.7898	0.9099	9.9190	0.9148	9.8480

Figure 20. Dice Score and Hausdorff Distance for LV

For the Short Axis (SA) view, the model performs strongly with a Dice Score of 0.9218 for End-Diastole (ED) and 0.9398 for End-Systole (ES), indicating excellent overlap between predicted and ground truth masks. The Hausdorff Distances for SA are relatively low, with 4.93 mm for ED and 5.89 mm for ES, signifying good boundary accuracy and close alignment.

In the Long Axis (LA) view, the model maintains high performance in terms of Dice Score, with 0.9238 for ED and 0.9099 for ES, reflecting strong overlap. However, the Hausdorff Distances are notably higher in this view, at 9.79 mm for ED and 9.91 mm for ES, indicating that while the overall shape is well captured, boundary precision could be improved.

axis	MYO_ED_DC	MYO_ED_HD	MYO_ES_DC	MYO_ES_HD	MYO_DC	MYO_HD
SA	0.8152	7.3715	0.9013	7.6855	0.8583	7.5285
LA	0.8203	9.1992	0.9064	9.5132	0.8634	9.3563

Figure 21. Dice Score and Hausdorff Distance for MYO

For the Short Axis (SA) view, the model shows decent performance with a Dice Score of 0.8478 for End-Diastole (ED) and 0.8645 for End-Systole (ES), indicating reasonable overlap between predictions and ground truth. The Hausdorff Distances for SA are moderate, with 7.42 mm for ED and 7.61 mm for ES, reflecting acceptable boundary alignment but with some room for refinement.

Table 21 also represents results for Long Axis slices. Initially, DANN model resulted in non-converged low Dice Score, and huge Hausdorff Distance. After precise inspection and investigation of possible root of issue, we have concluded that static value for coefficient of Gradient Reversal Layer ($\lambda = 0.5$ in initial model) results poor segmentation of Myocardium. To address this issue, we decided to adjust the GRL coefficient dynamically based on the progress of the training using sigmoid-like function.

$$\lambda = \frac{2.0}{1.0 + e^{-10.0p}} - 1.0$$

where p is

$$p = \frac{i + (epoch * DataSet_{size})}{EPOCHS * DataSet_{size}}$$

Here, ***epoch*** is current epoch, ***i*** is current iteration, ***EPOCHS*** is the number of epochs overall (250 in our case), $\boldsymbol{DataSet_{size}}$ is the size of the data set.

Early in the training, the coefficient is close to 0, allowing the network to focus more on the classification task. As training progresses, the coefficient increases, making domain adaptation (via the GRL) stronger.

By this modification, we were able to improve the MYO Dice Score, and Hausdorff, which were previously extremely poor.

5.4 Comparison of Models

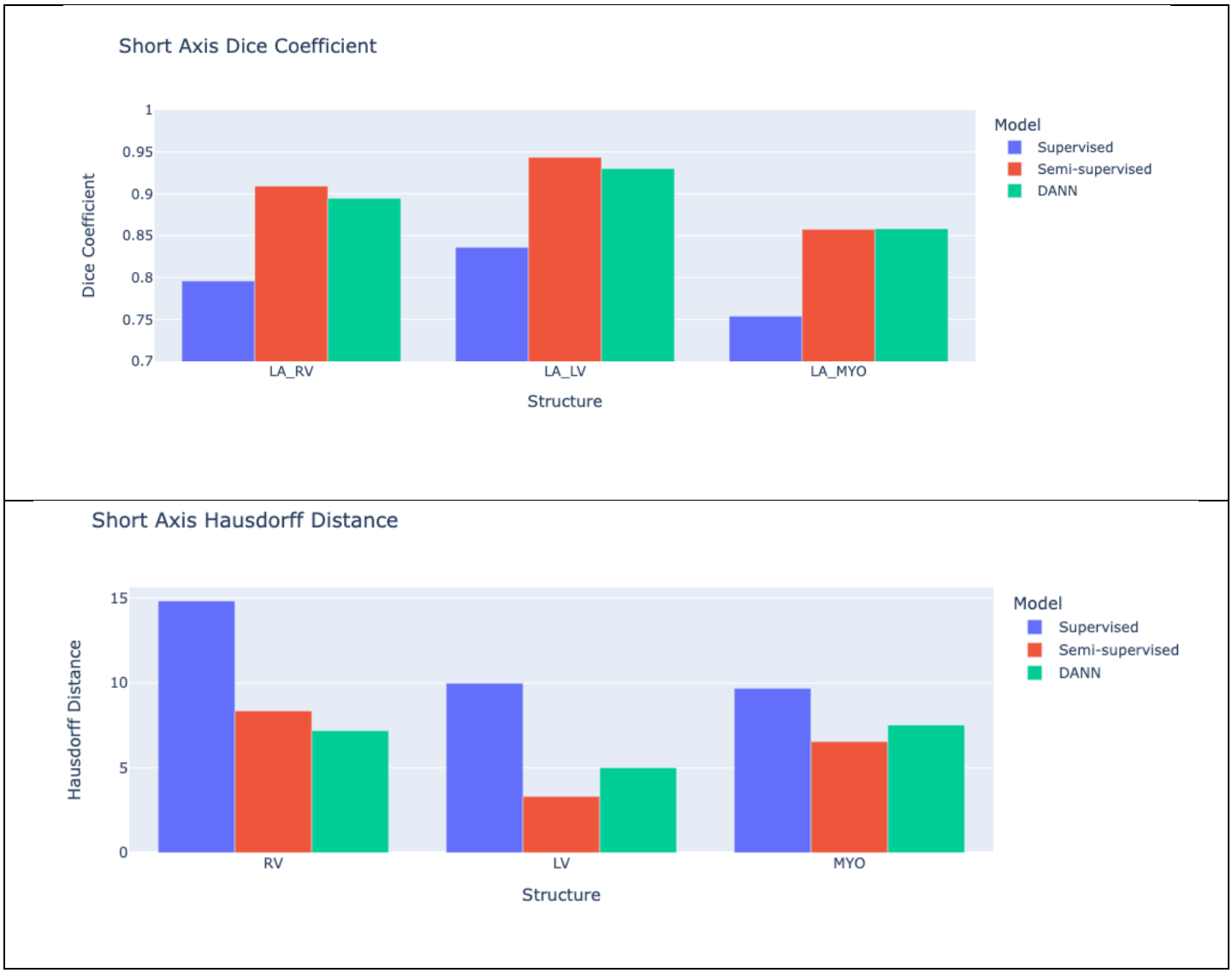


Figure 22. Comparison in cardiac structure metrics between Supervised, Semi-supervised model and DANN, Short-Axis

The two plots in Figure 22 present performance metrics for different models (Supervised, Semi-supervised, and DANN) across three structures (RV, LV, MYO). The first plot indicates that Semi-supervised model and DANN showed relatively simillar results. On the second plot DANN performed better on RV, while Semi-supervised model showed outstanding scores on LV and MYO. Supervised model overall showed slightly worse results than other candidates.

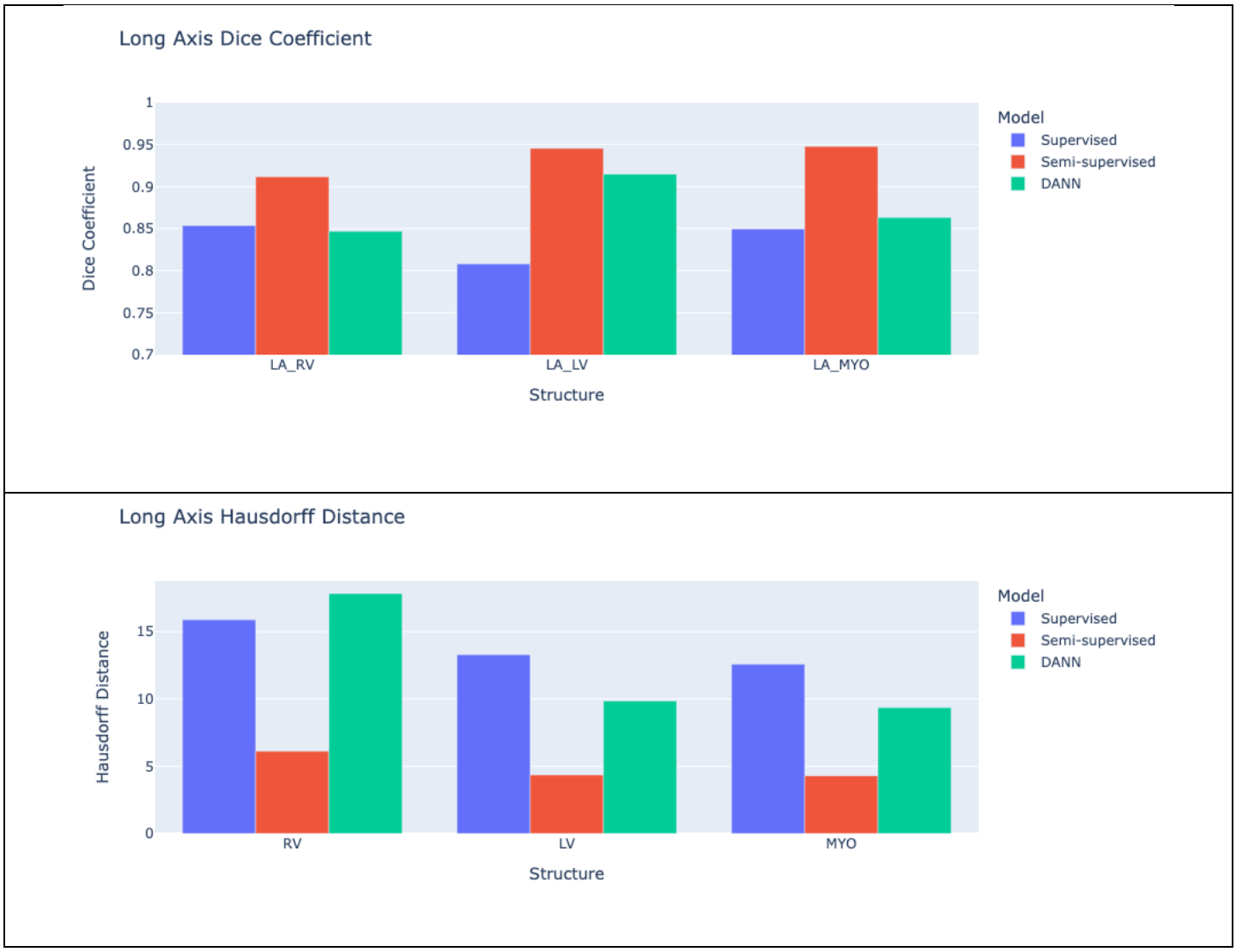


Figure 23. Comparison in cardiac structure metrics between Supervised, Semi-supervised model and DANN, Long-Axis

The comparison of the models' performance in cardiac structure metrics reveals notable differences in both the Dice Score and Hausdorff Distance. The Semi-supervised model consistently outperforms the DANN model and Supervised model in the RV and LV structures, achieving higher Dice Coefficient and low Hausdorff Distance, indicating better segmentation accuracy.

6. Production Deployment

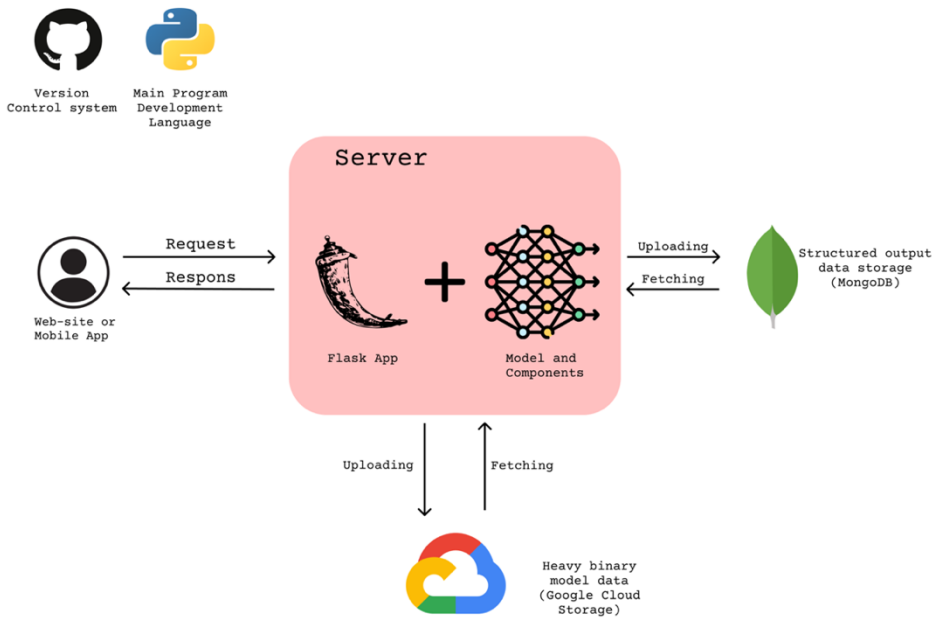
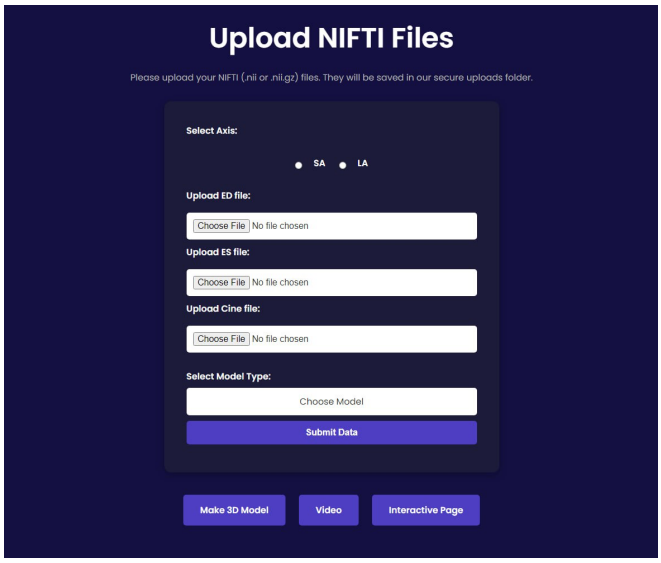


Figure 24. Web application workflow design

Our server is responsible for both segmentation model and backend site. After the user is authorized he can freely store and fetch all MRI images contained in Google Cloud Storage that he uploaded previously.



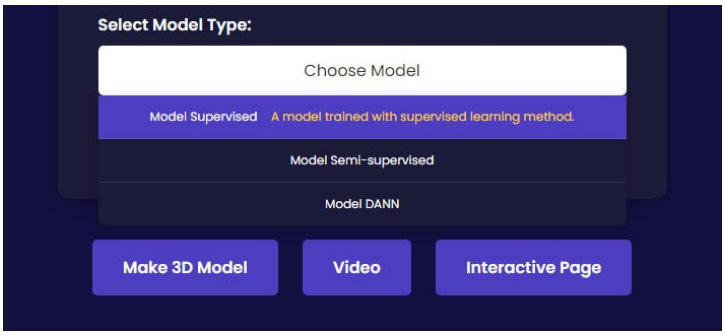


Figure 25. Upload windows

In the upload window (Figure 25), user should upload ED, ES and Cine files of Cardiac MRI. After that, user can choose which model he wants to use for segmentation.

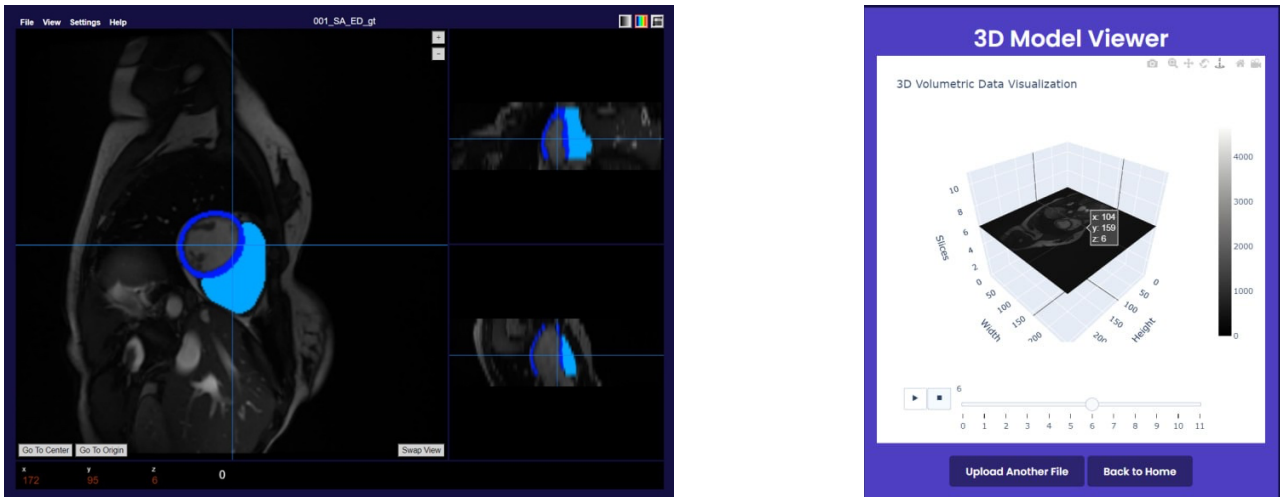


Figure 26. 3D Visualization and Segmentation Inspection tool

After processing MRI images, user can inspect it with tools like rescaling, measuring distance, and changing the colors of separate structures (LV, RV, MYO). Also, we included 3D Volumetric Data Visualization feature to inspect multi-sliced MRI data.

7. Updated Project Implementation Plan

7.1 Updated Development Schedule

Division	Development Schedule																	
	May		June				July				August				September			
	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. Background and Problem Research Review existing works on heart MRI segmentation, focusing on machine learning (ML) and deep learning (DL) methods, such as U-Net, which has proven effective in medical image segmentation.																		

Recognize the need for a reliable, robustness deep learning-based system to automate segmentation and simplify the interpretation of MRI data, addressing both clinical and technical needs.																	
2. Solution Suggestion and Development Suggest the development of a website that integrates a deep learning-based heart MRI segmentation model, allowing users to upload NIFTI files and get segmentation results. Decide to use Python (Flask for web framework), PyTorch for model integration, and NIFTI format support for MRI data. Design the system in a modular way, allowing separation of the model training and prediction functionalities from the website's user interface and file management components.																	
3. Finding Supplementary Materials: Dataset, Code Snippets Identify and acquired relevant datasets like the M&M2 Challenge dataset, containing MRI images of the heart with ground truth segmentations. Search for publicly available code snippets related to MRI preprocessing, U-Net model																	

implementation, and file handling in Flask to expedite development. Select libraries such as NiBabel for handling NIFTI file formats and preprocessing MRI data.																	
4. Data Preparation, Data Visualization, and Preprocessing Ensure all MRI scans were in NIFTI format and contained proper annotations for supervised training. Apply data augmentation techniques (rotation, zooming, flipping) to artificially expand the dataset and avoid overfitting. Use Matplotlib and other tools to visualize MRI slices and corresponding segmentations, ensuring the integrity of the data. Normalize MRI data to ensure uniform input for the model and resized images for efficient training.																	
5. Model Design and Development Design a semi-supervised learning framework incorporating quality control (QC) and candidate selection, allowing the model to leverage both labelled and unlabelled MRI data for segmentation. The framework enhances																	

segmentation accuracy by using candidate selection for refining predictions.																	
6. Model Training, Testing, and Fine-tuning Train the semi-supervised model using labelled MRI data along with a portion of unlabelled data, employing candidate selection during training for refining predictions and improving segmentation. Test the model on unseen MRI data, focusing on evaluating segmentation performance and minimizing overfitting by assessing on both labelled and unlabelled data. Fine-tune the model by adjusting hyperparameters and incorporating regularization techniques to improve generalization.																	
7. Model Evaluation, and Results Representation Evaluate using metrics such as Dice Score, Hausdorff Distance (HD) for segmentation tasks, with particular emphasis on performance improvements from candidate selection. Compare results with benchmarks from the literature, demonstrating competitive performance, especially for left ventricle and myocardium segmentation, while showing the advantage of leveraging unlabeled data.																	

8. Documentation and Presentation Complete and submit an intermediate report detailing progress, methodologies employed, and preliminary findings.																	
9. Model Design and Development Implement the Domain-Adversarial Neural Network (DANN) architecture to address domain adaptation challenges, allowing the model to generalize better across different MRI datasets. The adversarial training strategy helps reduce the domain shift between labelled source and unlabelled target domains.																	
10. Model Training, Testing, and Fine-tuning Train the DANN model on labelled source MRI data while applying adversarial training to ensure proper alignment of features between source and target MRI data. A validation split was used to monitor performance during training. Evaluate the model on unseen MRI datasets from the target domain to verify the model's ability to adapt to new data sources while maintaining segmentation accuracy. Fine-tune the DANN model by optimizing hyperparameters and employing domain-specific augmentation strategies to enhance model robustness and prevent overfitting.																	

11. Model Evaluation, and Results Representation Assess using the same metrics - Dice Score, and HD - with additional evaluation focusing on the model's ability to generalize across different domains. Compare results with baseline models, highlighting the improvements in domain adaptation and generalization across diverse MRI datasets, showing competitive performance in segmentation tasks.																	
12. Post-processing and Results Visualization Apply smoothing and morphological operations to the predicted segmentation masks to remove noise and enhance the visual output. Visualize predicted segmentation masks alongside ground truth using overlay techniques to illustrate the model's accuracy.																	
13. Application Design Development Create wireframes and workflows, focusing on ease of use. The design allowed users to upload MRI data, view previous uploads, and download segmented images.																	

Implement security features such as password protection, secure file handling, and account management (password updates, deletions). Integrate the trained model into the Flask application, ensuring real-time interaction where users can upload their MRI files and receive the segmented results.																	
14. Application Implementation Implement core functionalities, including: o **NIFTI file upload:** Enable users to upload MRI files. o **Profile management:** Allow users to manage their profiles and update passwords. o **Model predictions:** Integrate the U-Net model to generate segmentation results and display them back to the user. Enable users to view and download previously uploaded MRI files and results. Use Flask to handle requests, manage file storage, and process MRI data in the background. Develop an intuitive front-end for file uploads, results viewing, and user account interactions.																	
15. Documentation and Presentation																	

Document the entire project process, including the problem background, model development, evaluation, and results analysis, and implementation details. Create a concise poster and presentation summarizing the project, including key results, diagrams of the U-Net architecture, and visual examples of MRI segmentation. Prepare the final codebase, evaluation table, demonstration plan for submission, ensuring the project is reproducible and easy to follow.																	

7.2 Role Division

Name	Role
이슬람 살리흐	**1. Data Preparation, Data Visualization, and Pre-processing:** **Data visualization:** Visualized MRI slices and segmentation results to check data integrity. **Pre-processing steps:** Normalized MRI data and resizing images for efficient model training. **2. Post-processing and Results Visualization:** **Post-processing:** Applied smoothing and morphological operations to the predicted segmentation masks to remove noise and enhance the visual output.

	Results visualization: Visualized predicted segmentation masks alongside ground truth using overlay techniques to illustrate the model's accuracy. **3. Application Design Development:** **Model integration**: Integrated the trained model into the Flask application, ensuring real-time interaction where users can upload their MRI files and receive the segmented results. **4. Application Implementation** **Backend implementation**: Used Flask to handle requests, manage file storage, and process MRI data in the background. **Model predictions**: Integrated the U-Net model to generate segmentation results and display them back to the user.
케네스 예라슬	**1. Model Design and Development:** **Architecture design:** Researched, chose, and implemented the U-Net architecture, and Domain Adversarial Neural Network techniques. **2. Model Training, Testing, and Fine-tuning:** **Model training:** Handled the training process of the U-Net on the MRI dataset with validation splits. **Testing:** Evaluated segmentation performance on unseen MRI data to ensure the model is generalizing well. **Fine-tuning:** Adjusted hyperparameters and added dropout layers to improve the model's generalization and reduce overfitting. **3. Application Design Development:**

	Security considerations: Implemented security features such as password protection, secure file handling, and account management (password updates, deletions). **4. Application Implementation:** **Model integration:** Integrated model for inference and production purposes. **NIFTI file upload**: Enabled users to upload MRI files. **File management**: Enabled users to view and download previously uploaded MRI files and results.
누가예바 알트나이	**1. Data Preparation, Data Visualization, and Pre-processing:** **Data cleaning:** Ensured that MRI scans are in NIFTI format and contain proper annotations. **Data augmentation:** Expanded the dataset through techniques like rotation, zooming, and flipping. **2. Model Evaluation and Results Representation:** **Evaluation metrics:** Calculated and analysed metrics like Dice Score, Hausdorff Distance (HD), and accuracy. **Results representation:** Compared the model's results with benchmarks in literature for detailed evaluation. **3. Application Design Development:** **User flow design**: Created wireframes and workflows, focusing on ease of use. The design allowed users to upload MRI data, view previous uploads, and download segmented images.

	4. Application Implementation: **Frontend development:** Developed an intuitive front-end for file uploads, results viewing, and user account interactions. **Profile management:** Allowed users to manage their profiles and update passwords.
Common	**1. Background and Problem Research:** **Problem Research and Literature Review:** Reviewed existing works on heart MRI segmentation, focusing on machine learning (ML) and deep learning (DL) methods, such as U-Net, which has proven effective in medical image segmentation. **Problem Identification:** Recognized the need for a reliable, robustness deep learning-based system to automate segmentation and simplify the interpretation of MRI data, addressing both clinical and technical needs. **2. Solution Suggestion and Development:** **Proposed solution:** Suggested the development of a website that integrates a deep learning-based heart MRI segmentation model, allowing users to upload NIFTI files and get segmentation results. **Technology stack:** Decided to use Python (Flask for web framework), PyTorch for model integration, and NIFTI format support for MRI data. **Modular approach:** Designed the system in a modular way, allowing separation of the model training and prediction functionalities from the website's user interface and file management components.

	3. Finding Supplementary Materials: Dataset, Code Snippets: **Dataset search:** Identified and acquired relevant datasets like the M&M 2 Challenge dataset, containing MRI images of the heart with ground truth segmentations. **Code snippets:** Searched for publicly available code snippets related to MRI pre-processing, U-Net model implementation, and file handling in Flask to accelerate the development process. **Pre-processing tools:** Selected libraries such as NiBabel for handling NIFTI file formats and pre-processing MRI data. **4. Report Writing, Poster, and Supplementary Materials:** **Final report:** Documented the entire project process, including the problem background, objectives, model development, model training, evaluation, results analysis, and implementation details. **Poster creation:** Created a concise poster and presentation summarizing the project, including key results, diagrams of the U-Net architecture, and visual examples of MRI segmentation. **Supplementary materials:** Prepared the final codebase, evaluation table, demonstration plan for submission, ensuring the project is reproducible and easy to follow.

8. Conclusion

8.1 Results Discussion

In conclusion, this study successfully demonstrated the efficacy of combining Semi-Supervised Learning with QC Candidate Selection and the U-Net based Domain-Adversarial Neural Network (DANN) in enhancing the generalizability of cardiac MRI segmentation models across diverse datasets. The Semi-Supervised Learning

approach yielded satisfactory results in segmenting critical cardiac structures, including the Left Ventricle (LV), Right Ventricle (RV), and Myocardium (MYO), across both Short Axis (SA) and Long Axis (LA) views. Notably, the DANN model excelled in the SA segmentations, particularly in terms of the Hausdorff Distance metric, outperforming the Semi-Supervised model significantly. Overall, comparing to traditional supervised learning method, both models showed outstanding results in diverse dataset segmentation.

8.2 Future Work

In this research we worked with only one type of domain adaptation technique, however there are many peculiar methods that are worth investigating. Such methods include Correlation Alignment (CORAL) for Unsupervised Domain Adaptation, Adversarial Discriminative Domain Adaptation (ADDA). CORAL, for instance, mitigates domain shift by aligning the second-order statistics of both the source and target distributions, all while not needing any target labels. On the other hand, ADDA initially develops a discriminative representation based on the labels from the source domain then creates a distinct encoding that aligns the target data with this representation by employing an asymmetric mapping, which is optimized through a domain-adversarial loss.

In addition, diversifying the dataset with more vendor types, scanner models and clinical centers might further improve the ability of models to generalize.

References

〈단행본 예시〉

[1] M. I. Razzak, S. Naz, and A. Zaib. *Deep Learning for Medical Image Processing: Overview, Challenges and the Future*, in N. Dey, A. Ashour, and S. Borra Ed., Classification in BioApps, Lecture Notes in Computational Vision and Biomechanics, Vol. 26, Springer, Cham, 2018.

〈논문지 예시〉

[1] V. M. Campello, P. Gkontra, C. Izquierdo, C. Martin-Isla, A. Sojoudi, P. M. Full, K. Maier-Hein, Y. Zhang, Z. He, J. Ma, M. Parreno, A. Albiol, F. Kong, S. C. Shadden, J. C. Acero, V. Sundaresan, M. Saber, M. Elattar, H. Li, and K. Lekadir, "Multi-Centre, multi-vendor and multi-disease cardiac segmentation: The M&MS Challenge," *IEEE Transactions on Medical Imaging*, Vol. 40, No. 12, pp. 3543-3554, Dec. 2021.

[2] D. A. Bluemke, L. Moy, M. A. Bredella, B. B. Ertl-Wagner, K. J. Fowler, V. J. Goh, E. F. Halpern, C. P. Hess, M. L. Schiebler, and C. R. Weiss, "Assessing Radiology Research on Artificial Intelligence: A Brief Guide for Authors,

Reviewers, and Readers-From the Radiology Editorial Board," *Radiology*, Vol. 294, No. 3, pp. 487-489, Mar. 2020.

[3] S. M. McKinney, M. Sieniek, V. Godbole, J. Godwin, N. Antropova, H. Ashrafian, T. Back, M. Chesus, G. S. Corrado, A. Darzi, M. Etemadi, F. Garcia-Vicente, F. J. Gilbert, M. Halling-Brown, D. Hassabis, S. Jansen, A. Karthikesalingam, C. J. Kelly, D. King, J. R. Ledsam, and S. Shetty, "International evaluation of an AI system for breast cancer screening," *Nature*, Vol. 577, No. 7788, pp. 89-94, Jan. 2020.

[4] H. Guan and M. Liu, "Domain Adaptation for Medical Image Analysis: A Survey," *IEEE Transactions on Biomedical Engineering*, Vol. 69, No. 3, pp. 1173-1185, Mar. 2022.

[5] P. Goel and A. Ganatra, "Unsupervised Domain Adaptation for Image Classification and Object Detection Using Guided Transfer Learning Approach and JS Divergence," *Sensors*, Vol. 23, No. 9, pp. 4436, 2023.

〈학술대회 발표 논문집 예시〉

[1] J. Corral Acero, V. Sundaresan, N. Dinsdale, V. Grau, and M. Jenkinson, "A 2-step deep learning method with domain adaptation for multi-centre, multi-vendor and multi-disease cardiac magnetic resonance segmentation," *Proc. of Lecture Notes in Computer Science*, Vol. 12592, pp. 196-207, 2021.

[2] L. Li, W. Ding, L. Huang, and X. Zhuang, "Right ventricular segmentation from short- and long-axis MRIs via information transition," *Proc. of Lecture Notes in Computer Science*, pp. 259-267, 2022.

[3] R. P. Poudel, P. Lamata, and G. Montana, "Recurrent fully convolutional neural networks for multi-slice MRI cardiac segmentation," *Proc. of Lecture Notes in Computer Science*, pp. 83-94, 2017.

[4] M. Paschali, S. Conjeti, F. Navarro, and N. Navab, "Generalizability vs. Robustness: Investigating Medical Imaging Networks Using Adversarial Examples," *Proc. of Medical Image Computing and Computer Assisted Intervention - MICCAI 2018, Lecture Notes in Computer Science*, Vol. 11070, pp. 56-64, 2018.

[5] M. Beetz, J. C. Acero, and V. Grau, "A Multi-view Crossover Attention U-Net Cascade with Fourier Domain Adaptation for Multi-domain Cardiac MRI Segmentation," *Proc. of Lecture Notes in Computer Science*, pp. 323-334, 2022.

[6] F. Galati, M. A. Zuluaga, "Using Out-of-Distribution Detection for Model Refinement in Cardiac Image Segmentation," *Proc. of Statistical Atlases and Computational Models of the Heart, STACOM 2021, Lecture Notes in Computer Science*, vol. 13131, pp. 374-382, Springer, Cham, 2022.

[7] O. Ronneberger, P. Fischer, T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Proc. of the Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015, Lecture Notes in*

Computer Science, vol. 9351, pp. 234-241, Springer, Cham, 2015.

[8] F. Galati, M. A. Zuluaga, "Efficient Model Monitoring for Quality Control in Cardiac Image Segmentation," *Proc. of the Functional Imaging and Modeling of the Heart (FIMH 2021), Lecture Notes in Computer Science*, vol. 12738, pp. 101-111, Springer, Cham, 2021.

[9] P. Bergmann, S. Löwe, M. Fauser, D. Sattlegger, and C. Steger, "Improving unsupervised defect segmentation by applying structural similarity to autoencoders," *Proc. of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, pp. 372-380, 2019.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *Proc. of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778, 2016.

〈WEB Site 예시〉

[1] B. Glocker, R. Robinson, D. C. Castro, Q. Dou, and E. Konukoglu. (2019, October 10). Machine learning with multi-site imaging data: An empirical study on the impact of scanner effects [Online]. Available: https://arxiv.org/abs/1910.04597

[2] P. V. Tran. (2017, April 27). A fully convolutional neural network for cardiac segmentation in short-axis MRI [Online]. Available: https://arxiv.org/abs/1604.00494

[3] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. March, and V. Lempitsky. (2016). Domain-Adversarial training of neural networks [Online]. Available: https://jmlr.org/papers/v17/15-239.html