TECHNICAL UNIVERSITY OF CRETE

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

Analysis of Cardiac Anatomy Biomedical Images with the use of Vision Transformers

Committee

Professor Michalis E. Zervakis (Supervisor)

Professor Dionysios Christopoulos

Professor Thrasyvoulos Spyropoulos

Author

Stelina Naka

A thesis submitted in fulfillment of the requirements
for the diploma of Electrical and Computer Engineer

Abstract

The heart is one of the most complex organs of human body with multiple substructures and the anatomy of the whole heart is a basic requirement for the developing of many clinical applications. To study spatially heart function Magnetic Resonance Imaging (MRI) and Computed Tomography (CT) are the most common ways. In this regard, the whole-heart segmentation is vital in medical imaging analysis, providing the potential for diagnosis and treatment options of the Cardiovascular Diseases (CVD). However, the automated segmentation can be challenging due to variation of the heart shape. In this thesis, an enhanced method based on the insights of the MICCAI Multi-Modality Whole Heart Segmentation (MM-WHS) challenge evaluations is proposed. The challenge provides a dataset of 20 MRI and 20 CT volumes and their manually segmented labels. For later-on model training, most of the automated segmentation tasks on medical images are based on Convolutional Neural Networks, a hybrid Vision Transformer (ViT) model is introduced in this thesis. The so-called *'ViTSegment'*, the proposed model, is a Vision Transformer-based encoder for capturing long range dependencies and a convolutional decoder for accurate boundary detection. The proposed algorithm was trained and evaluated on the dataset from the challenge. Due to the low number of data, we further proceeded with data augmentation techniques to expand the dataset. On CT dataset it exhibited a better dice score of 92.65 $\pm$ 2.17% compared to the MRI dataset (91.50 $\pm$ 1.72%). To boost the results of the evaluation, a comparative analysis was implemented between the ViTSegment, U-Net and UNETR models. The ViTSegment outperforms the other two models, with U-Net achieving a dice score of 82.67 $\pm$ 8.70% on the CT dataset and 81.30 $\pm$ 5.47% on the MRI, while UNETR scores 86.33 $\pm$ 0.74% for CT and 84.94 $\pm$ 6.25% for MRI, highlighting its robustness and efficiency. ViTSegment model shows essential potential that is paving the way for robust automated whole-heart segmentation in medical image analysis.

Περίληψη

Η καρδιά είναι ένα από τα πιο πολύπλοκα όργανα του ανθρώπινου σώματος με πολλαπλά μέρη και η ανατομία ολόκληρης της καρδιάς αποτελεί βασική προϋπόθεση για την ανάπτυξη πολλών κλινικών εφαρμογών. Για τη μελέτη της λειτουργίας της καρδιάς σε χωρικό επίπεδο, η Μαγνητική Τομογραφία (MRI) και η Αξονική Τομογραφία (CT) είναι οι πιο συνηθισμένες μέθοδοι. Στο πλαίσιο αυτό, η κατάτμηση ολόκληρης της καρδιάς είναι ζωτικής σημασίας στην ανάλυση ιατρικών εικόνων, παρέχοντας δυνατότητες για διάγνωση και την επιλογή θεραπείας των Καρδιαγγειακών Νοσημάτων (CVD). Ωστόσο, η αυτοματοποιημένη κατάτμηση μπορεί να γίνει απαιτητική λόγω της παραλλαγής στο σχήμα της καρδιάς από άνθρωπο σε άνθρωπο. Σε αυτή την μελέτη, προτείνεται μια ενισχυμένη μέθοδος βασισμένη στις γνώσεις που προέκυψαν από τα ευρύματα των αξιολογήσεων του διαγωνισμού MICCAI Multi-Modality Whole Heart Segmentation (MM-WHS). Ο διαγωνισμός παρέχει ένα σύνολο δεδομένων που περιλαμβάνει 20 MRI και 20 CT τρισδιάστατες εικόνες και τις χειροκίνητα κατατετμημένες ετικέτες τους. Ενώ οι περισσότερες από τις αυτοματοποιημένα μοντέλα κατάτμησης ιατρικών εικόνων βασίζονται σε Convolutional Neural Networks (CNN), στην διατριβή αυτή παρουσιάζεται ένα υβριδικό μοντέλο Vision Tranformer (ViT). Το προτεινόμενο μοντέλο, με όνομα *'ViTSegment'*, είναι ένας κωδικοποιητής βασισμένος σε Vision Transformer για την καταγραφή εξαρτήσεων μεγάλης κλίμακας και ένας αποκωδικοποιητής συνελικτικού τύπου για την ακριβή ανίχνευση των ορίων. Ο προτεινόμενος αλγόριθμος εκπαιδεύτηκε και αξιολογήθηκε στο σύνολο δεδομένων του διαγωνισμού. Λόγω του μικρού αριθμού δεδομένων, προχωρήσαμε περαιτέρω με τεχνικές αύξησης δεδομένων (data augmentation) για την επέκταση του συνόλου δεδομένων. Στο σύνολο δεδομένων CT παρουσίασε καλύτερο Dice score $92.65 \pm 2.17\%$ σε σύγκριση με το MRI σύνολο ($91.50 \pm 1.72\%$). Για τη βελτίωση των αποτελεσμάτων της αξιολόγησης, πραγματοποιήθηκε συγκριτική ανάλυση μεταξύ των μοντέλων ViTSegment, U-Net και UNETR. Το ViTSegment υπερέχει των άλλων δύο μοντέλων, με το U-Net να επιτυγχάνει Dice score $82.67 \pm 8.70\%$ στο σύνολο CT και $81.30 \pm 5.47\%$ στο MRI, ενώ το UNETR σημειώνει $86.33 \pm 0.74\%$ για το CT και $84.94 \pm 6.25\%$ για το MRI, αναδεικνύοντας τη σταθερότητα και την αποδοτικότητα του. Το μοντέλο ViTSegment παρουσιάζει σημαντικές προοπτικές, ανοίγοντας τον δρόμο για αξιόπιστη αυτόματη κατάτμηση ολόκληρης της καρδιάς στην ανάλυση ιατρικών εικόνων.

Acknowledgments

I would like to thank my supervisor Professor Michalis Zervakis and also Dr. Marios Antonakakis for their support and guidance throughout this project. I would also like to thank the members of my thesis committee Professor Dionysios Christopoulos and Professor Thrasyvoulos Spyropoulos.

A special thanks to my family and friends for being by my side, supporting and encouraging me throughout this journey, despite all the challenges that I faced.

Contents

List of Figures

Chapter 1

Introduction

The heart, as the main organ of the cardiovascular system, is responsible for circulating blood throughout the human body. In this regard, even a brief dysfunction of its normal state can have serious consequences. Cardiovascular disease is currently the leading cause of death worldwide, this highlights the need for precise diagnosis and treatment. Therefore, efficient monitoring of its physiological state is essential, especially for patients with cardiovascular diseases (CVDs). With its intricate anatomy and critical role in overall health, the heart requires advanced imaging technologies and computational tools.

This study focuses in particular on the analysis and segmentation of heart anatomy, which is crucial in the treatment of cardiovascular diseases (CVDs). Both Magnetic Resonance Imaging (MRI) and Computed Tomography (CT) are imaging modalities offering clinicians a unique opportunity for getting detailed insights into morphological and functional characteristics of the human heart.

Cardiac image segmentation is an essential task in this process, serving as a first step for acquiring quantitative measurements. Especially in 3D data, manual segmentation is really time-consuming, prone to human error, and highly dependent on the expertise of the user. Therefore, the need for automated approaches has grown.

Automated segmentation, although promising, faces several critical challenges. Variations in heart anatomy across patients, unclear boundaries between substructures, and low image quality between different regions can pose significant challenges to accurate

segmentation. Additionally, issues like class imbalance and the need for large, annotated datasets, which are often limited in medical imaging, further complicate the process. Many traditional approaches have achieved success in segmentation but are constrained by the complexity of anatomical structures.

In deep learning, recent advancements with Vision Transformers (ViTs) have shown promise in medical image classification tasks. However, ViTs have seen limited application in 3D medical segmentation tasks.

In this thesis, a hybrid segmentation model, ViTSegment, is introduced that combines both Vision Transformer and convolutional architecture for cardiac image segmentation. ViTSegment is a hybrid model with a Vision Transformer-based encoder for capturing long range dependencies and a convolutional decoder for accurate boundary detection. By introducing Vision Transformers into the automated whole-heart segmentation task it is shown their potential to capture long-range dependencies in complex anatomical structures.

Evaluation and testing results indicate that the model achieves a great segmentation accuracy of most structures compared to two other regularly used models. To train and evaluate the proposed model and the other two mentioned, the dataset from MICCAI 2017 Multi-Modality Whole Heart Segmentation (MM-WHS) challenge is used. The dataset consists of 20 CT and 20 MRI volumes of the heart, that have been manually segmented. Given the dataset's limited size, data augmentation techniques are applied to enhance the model's performance as much as possible. Additionally, a hybrid Dice-Cross Entropy loss function is employed during the training process to reduce the issue of class imbalance.

1.1 Related Work

Due to the MICCAI challenge a lot of related work were reviewed from the participants' papers [50]. The participants applied various strategies. One of the traditional techniques for automatic whole-heart segmentation was the multi-atlas segmentation (MAS), with different methodologies being used. MAS-based algorithms exhibited anatomical realistic results. However, they were computationally expensive and they

struggled a lot because of the poor image quality especially in MRI dataset.

Several participants also employed deep learning CNN-based architectures with various implementations. Again, despite of the particular success on CT dataset, the performance on MRI wasn't good.

The MICCAI challenge provided insights on various algorithms, indicating their strengths and their limitations. The deep learning based algorithms outperformed the MAS-based ones. Although, there were still challenges in handling class imbalances and low resolution. These researches continue to contribute in the development of better automatic whole heart segmentation algorithms, to achieve more accurate results.

1.2 Structure of thesis

This thesis is organized into the following five chapters:

- Introduction: This chapter is the Introduction. It provides an overview of the research that will follow, mentioning the overall motivation of this study and the related work that was reviewed in the beginning.
- Theoretical Background: This is the second chapter, were an overview of the human heart, medical imaging and deep learning is given.
- Methodology: This is the third chapter, were the proposed segmentation model is presented and detailed. Also the dataset and some training implementations are mentioned.
- Evaluation and Experimental Results: In this chapter, the performance of the proposed model is assessed and a comparative analysis with models such as U-Net and UNETR.
- Conclusion: The thesis closes with a brief conclusion and some potential future works based on some limitations that were observed during the study.

Chapter 2

Theoretical Background

2.1 Overview of Human Heart

The cardiovascular system provides blood supply throughout the body and consists of the heart, arteries, veins, and capillaries. The main organ of the cardiovascular system, the heart , is responsible to supply oxygen and nutrients to tissues and remove carbon dioxide and other wastes by pumping blood [7]. It's located in the thoracic cavity, between the two lungs, slightly left of the midline and divided into four chambers [11, 41]:

- Upper left atrium
- Upper right atrium
- Lower left ventricle
- Lower right ventricle

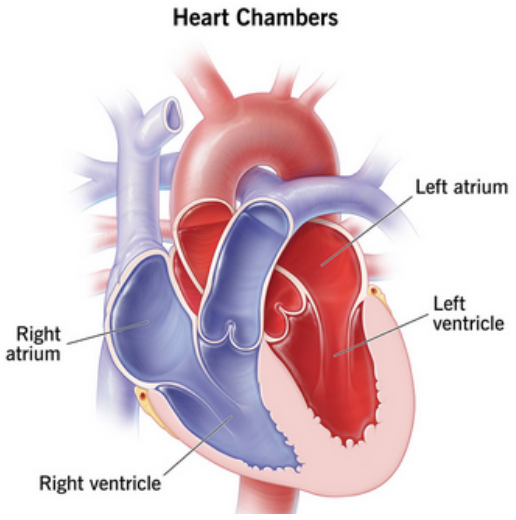

Figure 2.1: *The four chambers of the heart. [11]*

The atria receive blood returning to the heart and pump it into the ventricles, they on the other hand must pump the blood out of the heart to the lungs and the rest of the body so they have thicker walls due to their role. To regulate blood flow and ensure unidirectional movement, the heart contains four valves between the chambers ,the mitral, the aortic, the tricuspid and the pulmonary valve [7, 11, 41]. The heart walls consist of muscle layers that contract and relax in a coordinated rhythm to send blood throughout your body. To support this process, the blood vessels play also a critical role helping the heart transport blood [7]:

- **Arteries** (Aorta, Pulmonary): Carry blood away from heart.
- **Veins**: Transport the blood back to the heart.
- **Capillaries**: Allow the exchange of nutrients and wastes between the blood and body tissues.

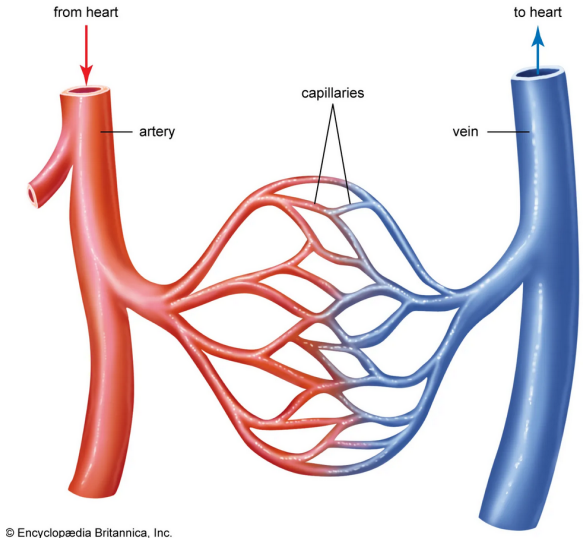

Figure 2.2: *The blood vessels of the heart. [4]*

The heart's pumping action is governed by the cardiac cycle, which consists of diastole and systole. During diastole the blood flows from the atria to the ventricles, the heart muscles relax and the heart fills with blood. In the systolic phase the heart muscles contract causing the blood to be ejected out of the chambers and into the aorta and pulmonary artery [11, 41]. An image of the heart is provided below for better visualization (Figure 2.3).

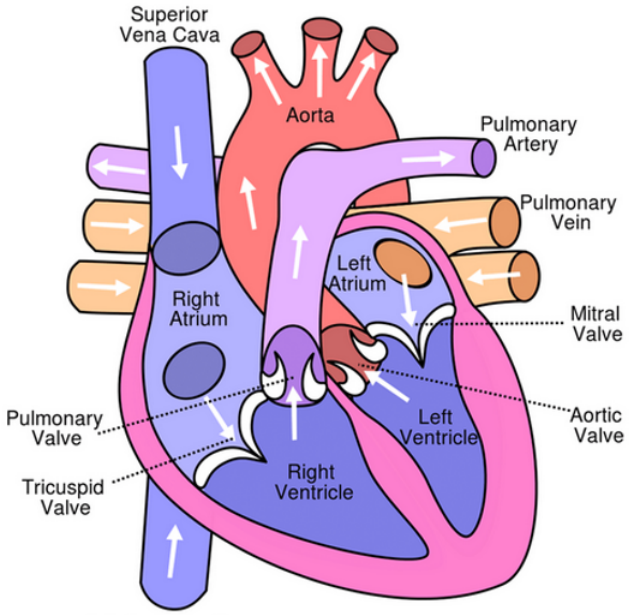

Figure 2.3: *Human Heart [43]*

2.2 Cardiac Imaging

Cardiac imaging plays a significant role in diagnosing cardiovascular conditions. The heart requires precise and detailed imaging to assess its structure and function. These techniques among others are Cardiac MRI and Cardiac CT, the first uses magnetic fields and radio waves providing us with high resolution images of the heart whereas Cardiac CT uses X-rays to obtain detailed images of the heart. Medical Imaging, in general, is a very active field in image analysis. Various image analysis methods are applied to medical images. Image analysis plays a critical role in transforming raw medical images into important information.

2.2.1 Image Segmentation

Image segmentation is a method used in cardiac imaging or medical imaging in general. It partitions an image into discrete groups of pixels with same characteristics, the image segments. This way the image becomes easier for the computer to analyse it and then

process it. The traditional image segmentation techniques resolve to annotations by analysing some of the pixels characteristics, like intensity, colour or brightness. On the other hand, the deep learning-based segmentation uses complex neural networks such as Convolutional Neural Networks (CNN) for advanced pattern recognition. The outputs in both approaches are the segmentation masks, that represent the specific delineation and shape of each object or feature, referred to as classes [53]. There are three types of image segmentation:

- **Semantic Segmentation** is the process of assigning class label to each pixel of an image, but doesn't provide context such as object boundaries.
- **Instance Segmentation** is the process of assigning a separate label to different instances of the same class in an image.
- **Panoptic Segmentation** is the process of labelling each pixel by class and also identifying different instances of the same class. It combines the advantages of both semantic and instance segmentation.

The following image (Figure 2.4) shows the difference between them.

(a) *Image* **(b)** *Semantic* **(c)** *Instance* **(d)** *Panoptic*

Figure 2.4: *Differences between the segmentation types. [27]*

2.2.2 Importance of Image Segmentation on Heart

According to World Health Organization (WHO), cardiovascular diseases are the main cause of death globally. Therefore, important steps have been made in cardiovascular research and clinical practice to try and improve early diagnosis and treatment of cardiac diseases or dysfunctions. Image segmentation plays a critical role in this, as it

helps isolate regions of interest (structures) and then healthcare professionals can accurately assess the heart conditions. Through image segmentation accurate extraction and precise interpretation of these anatomical information becomes easier. This has enabled the development of new application that aid in cardiology [8].

2.3 Introduction to Deep Learning

Deep learning is a subfield of machine learning that focuses on the use of multilayered neural networks to solve complex tasks. In additional with traditional machine learning models, deep learning models can extract features to make accurate outputs from raw data [8, 29, 33]. The multilayered neural networks are referred to as deep neural networks and can learn, recognize and classify objects and complex patterns within the data. The data pass through these layers in forward propagation, where each layer optimizes the prediction, and the model is trained by adjusting weights and biases through backpropagation to minimize prediction errors [29, 33]. This process requires a lot of computing power, typically provided by the GPU (Graphical Processing Unit). Deep learning can be "supervised", "semi-supervised", and "unsupervised", it can be used to solve regression, classification problems, segmentation and other problems.

2.3.1 Neural Networks

Deep learning revolves around neural networks, which are inspired by the functioning of the brain. Every neural network consists of artificial neurons and have three main layers, an input layer, one or more hidden layers, and an output layer. These layers work together to process the data [17, 25]. The artificial neuron is modelled by the biological neuron and has multiple inputs and one output. It consists of a linear part followed by a activation function, as shown in Figure 2.5. The output of an artificial neuron is:

$$f\left(\sum_{i=1}^{n} w_i x_i + b\right)$$

Where x_1,x_2,...,x_n are input values that are multiplied by their corresponding weights w_1,w_2,...,w_n and then added up together. After that a bias b is added to the result and finally the activation function, f is applied.

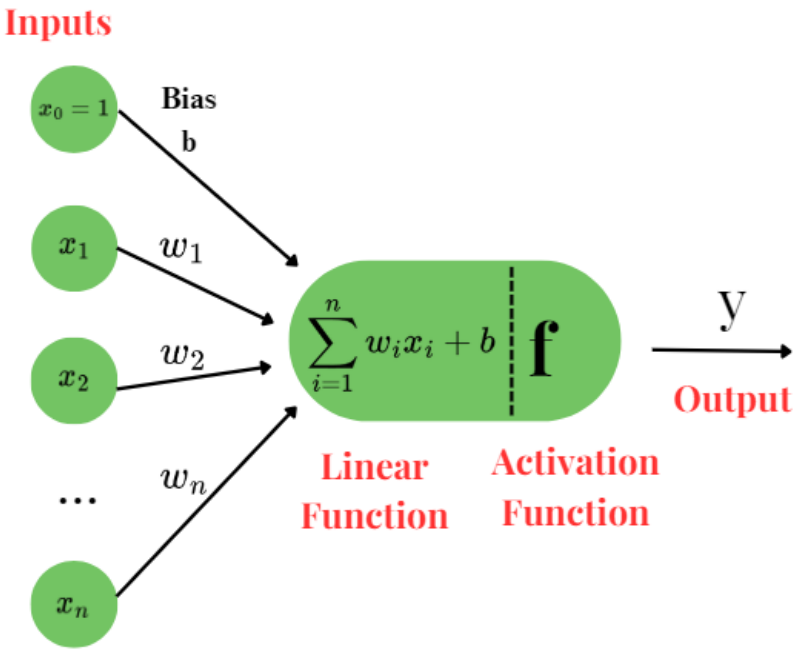

Figure 2.5: *Artificial Neuron*

The network's first layer passes the input into the hidden layer where each neuron extract information from the previous layer. Within the hidden layer, neurons compute a weighted sum of the inputs, apply a bias, and transform the result using an activation function. The outputs are transferred to the neurons in the next level. The activation function introduces non-linearity to the neuron's output, some of the activation functions used by artificial neural networks are the rectified linear unit (ReLU), softmax, sigmoid and other functions.

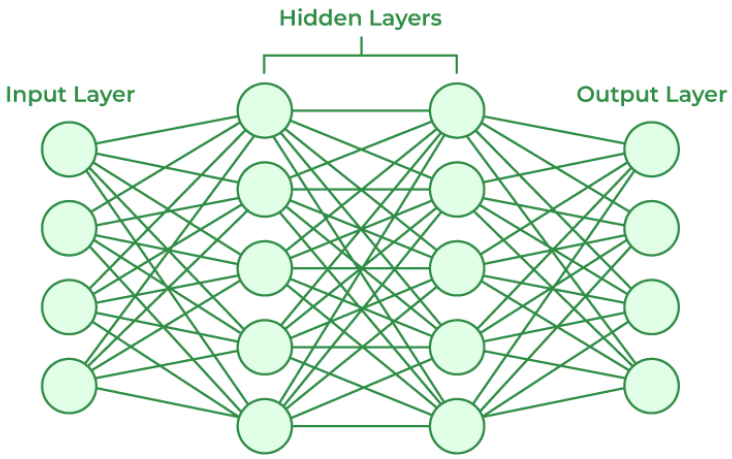

Figure 2.6: *Neural Networks Architecture [17]*

The process described above is repeated in all the hidden layers, with the extracted features progressively building on the output of the previous layer. At the final step, the output layer receives the processed data from the last hidden layer and produces the model predictions.

2.4 Fully Convolutional Network (FCN)

There are many different neural network models, the most popular one for image tasks is the Convolutional Neural Network (CNN) model. CNNs are equipped with convolutional layers, pooling layers, and fully connected layers, which work together and discover detailed patterns [2, 36]. They have proven particularly effective in recognizing and extracting features from visual data but they work with image patches, so a model that is applied to the entire image was introduced, FCN. Fully Convolutional Networks (FCNs) are a specialized type of artificial neural network (ANN) designed for image segmentation tasks. They are an extension of CNNs, where the fully connected layers are replaced by convolutional layers. This method makes them effective for pixel-wise segmentation and outputs segmentation masks directly. They follow an

encoder-decoder structure, where:

- The **encoder** extracts high-level features through convolution and pooling layers.
- The **decoder** reconstructs spatial details through upsampling (e.g., the transposed convolutions, see Figure 2.7) and convolution layers.

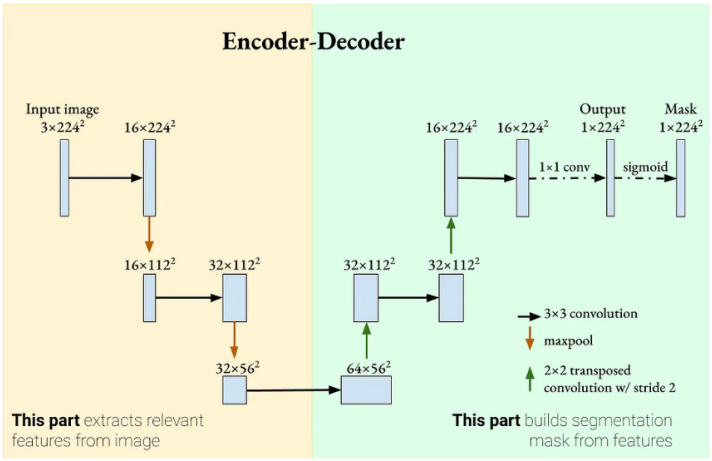

Figure 2.7: *Architecture of a Fully Convolutional neural Network (FCN). The FCN takes the image as input, extracts features via the encoder, and upsamples in the decoder (using transposed convolutions). In the end, it generates a segmentation map. [44]*

A way to achieve upsampling is by applying transposed convolution like in Figure 2.7. Although FCNs are a simple architecture, they have some limitations in capturing detailed information due to information being lost in the pooling layers. This limitation is resolved with skip connections between encoder and decoder that were introduced in U-Net architecture, a variant of FCNs for biomedical image segmentation[9, 37]. In this paper, the U-Net and FCN decoder architecture serves as inspiration for the proposed segmentation method to improve segmentation accuracy in cardiac imaging. More details on the implementation of this architecture will be provided in Chapter 3.

2.5 Vision Transformer

Transformers were first introduced by Vaswani et al. in the landmark paper "Attention is all you need" [46]. They are a type of deep learning model originally designed for

Natural Language Processing (NLP) tasks, such as machine translation, language modelling, and other. Unlike traditional deep learning models, they work with multi-head self-attention mechanisms and position-wise feed-forward networks instead of recurrence and convolutions. The previously mentioned mechanisms are what enable transformers to process inputs in parallel rather than sequentially, making them faster and more efficient. In Figure 2.8 the basic Transformer architecture is presented for NLP tasks.

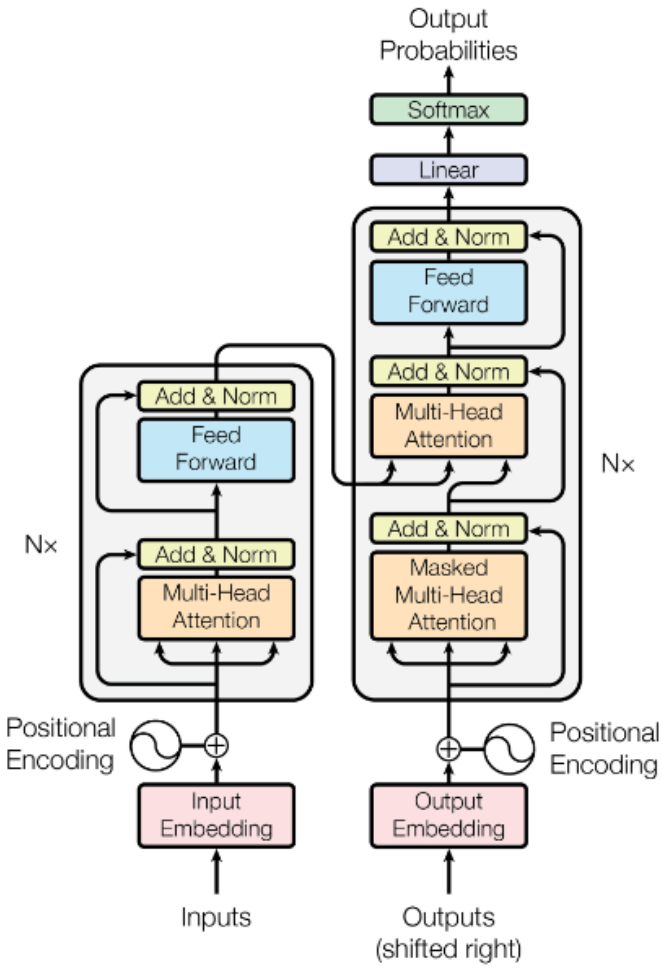

Figure 2.8: *The Transformer-model architecture [46]*

Inspired by the successful work of Transformers in NLP, a lot of adaptations have become for computer vision tasks, resulting in the Vision Transformers (ViT), introduced by Dosovitskiy et al. in "An Image is Worth 16x16 Words" [13]. The ability to cap-

ture global context is very important for vision tasks such as image segmentation and classification. The Vision Transformers process images as a sequence of patches, each image is divided into patches and is flatten and treated as tokens (words) same way as in NLP tasks. The patches are flattened and embedded into high dimensional vectors. To retain positional information the positional embedding are added before passing to the Transformer encoder. The model design of the transformer encoder was inspired by the original architecture in "Attention is All You Need" [46], it has multiple layers of multi-head self-attention (MHSA) and feed-forward networks (FFN). The model of Vision Transformer is illustrated in Figure 2.9, as an overview emphasizing on image patches and the encoder architecture.

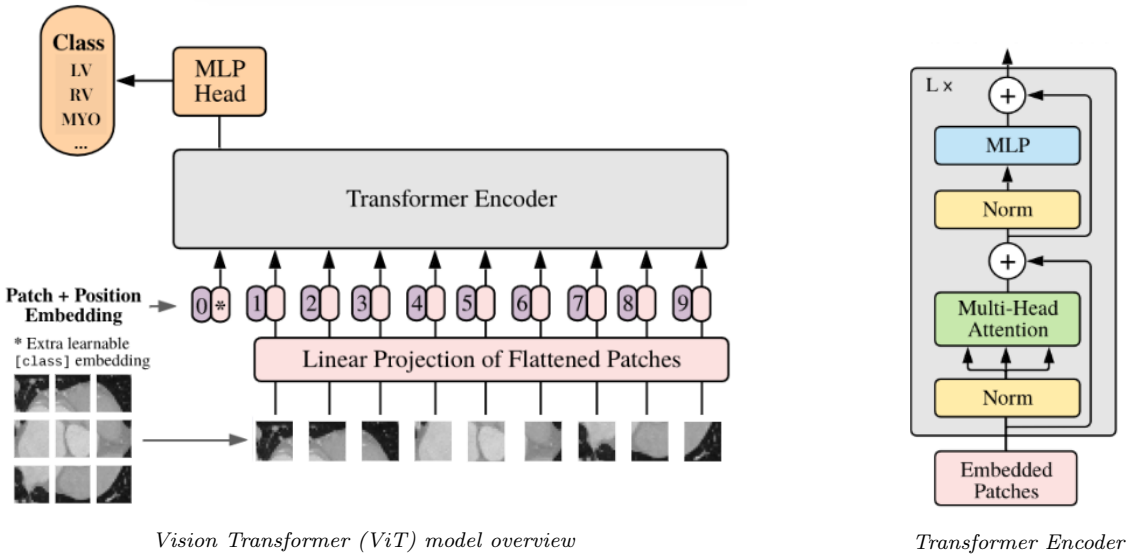

Figure 2.9: *Image is divided into fixed-size patches, linearly embedded into high-dimensional vectors, and combined with positional embeddings. The resulting sequence is passed through a standard Transformer encoder. [13]*

In comparison to state-of-the-art CNNs, the ViT offers computational efficiency, better performance and better resource usage. Based on the tests referred in the paper [13] ViT surpass CNNs, especially on diverse datasets. They achieve high performance with reduced computational costs and they demonstrate a strong performance even when trained on small datasets. That indicate scalability and flexibility to resource-constrained projects. While originally designed for image classification tasks, ViTs have

been used for segmentation tasks by being combined with CNNs. Table 2.1 provides a side-by-side comparison of Vision Transformers and Convolutional Neural Networks, highlighting their architectural differences, efficiency, and other.

Feature	**Vision Transformer (ViT)**	**Convolutional Neural Networks (CNNs)**
Core Mechanism	Multi-head Self-Attention	Convolutions
Efficiency	Scalable, works well with both large and small datasets	Efficient with smaller datasets
Performance	Better on large, diverse datasets	Strong on local features
Training Time	Slower with large datasets, complex mechanisms	Faster, but limited on large datasets
Resource Usage	More memory-intensive, but efficient with large data	Less memory-intensive, scales poorly with data size

Table 2.1: *Comparison of Vision Transformer (ViT) and Convolutional Neural Networks (CNNs)*

Although this section provides an overview of Vision Transformers and their advantages compared to CNNs, a more detailed explanation of their components is presented in Chapter 3.

Chapter 3

Methodology

This chapter outlines the methodology developed to create an automated approach for segmenting multi-modality images. The process can be really challenging due to the following reasons.[52]

- Small data size: 20 CT and 20 MRI training images.
- Large shape variations: The heart's anatomy can vary from patient to patient also the shape of the heart varies through the cardiac cycle as it contracts and relaxes.
- Indistinct boundaries: The different substructures have unclear boundaries, complicating the segmentation.
- Low image quality: This issue primarily pertains to MRI images that suffer from low resolution or noise.

3.1 Software Framework

The framework that was used for this study is **MONAI** (Medical Open Network for Artificial Intelligence)[5]. MONAI is an open-source project, PyTorch-based framework designed for deep learning in medical imaging. There were many tutorials on GitHub

about preprocessing, training and validating various datasets with different neural networks that proved invaluable.

3.2 Dataset and Preprocessing

3.2.1 Dataset

The dataset used in this study is from MM-WHS (Multi-Modality Whole Heart Segmentation) challenge [50, 52]. The challenge provided 120 3-dimensional cardiac images covering the whole heart, including 60 MRI and 60 CT volumes from which 20 were for training and validation (with label) and 40 for testing (without label).The training and validation volumes had a fixed shape of $[96 \times 80 \times 96]$, whereas the shape of the testing volumes varied. All the data cover the whole heart from the upper abdomen to the aortic arch and were given as a NIfTI format. The CT data were obtained from two state-of-the-art 64-slice CT scanners (Philips Medical Systems, Netherlands) using routine cardiac CT angiography protocol at two sites affiliated to Shanghai Shuguang Hospital. The in-plane resolution of the axial slices is 0.78×0.78 mm, and the average slice thickness is 1.60 mm. The cardiac MRI data were acquired from two hospitals in London, using 3D balanced steady state free precession (b-SSFP) sequences. The data were acquired at a resolution of around 2 mm, and reconstructed to half of its acquisition resolution, about 1 mm [50]. Manual labelling was provided for the seven substructures of the heart, that was done slice by slice from clinicians or students who were familiar with the whole heart anatomy. These are:

1. The Left Ventricle blood cavity, with label value 500.
2. The Right Ventricle blood cavity, with label value 600.
3. The Left Atrium blood cavity, with label value 420.
4. The Right Atrium blood cavity, with label value 550.
5. The Myocardium of the left ventricle, with label 205.
6. The Ascending Aorta, with label 820.

7. The Pulmonary Artery, with label 850.

The labels have been adjust for convenience, and the numbering corresponds to the enumeration above (from 1 to 7). The CT and MRI datasets were processed and trained separately to account for differences in modality-specific features and imaging characteristics.

3.2.2 Preprocessing

A preprocess pipeline was applied for both CT and MRI datasets to ensure consistency, prepare and optimize the input data. Some of MONAI's functions, such as Compose, helped to transform the data. First, each image and its corresponding label were cropped to remove irrelevant background, focusing on the heart region. Next, standardize the image orientation to RAS (Right-Anterior-Superior) format and resample the image to a consistent voxel size to ensure uniformity. Intensity normalization was also performed to enhance contrast and standardize brightness levels, improving the model's ability to differentiate between tissues. Then, padding was applied to ensure the images are divisible by 16,to accommodate the Vision Transformer model, which processes images in patches. Finally, each image and label are center-cropped to the region of interest, allowing the model's attention to focus on the heart. The preprocessing pipeline is shown at Figure 3.1.

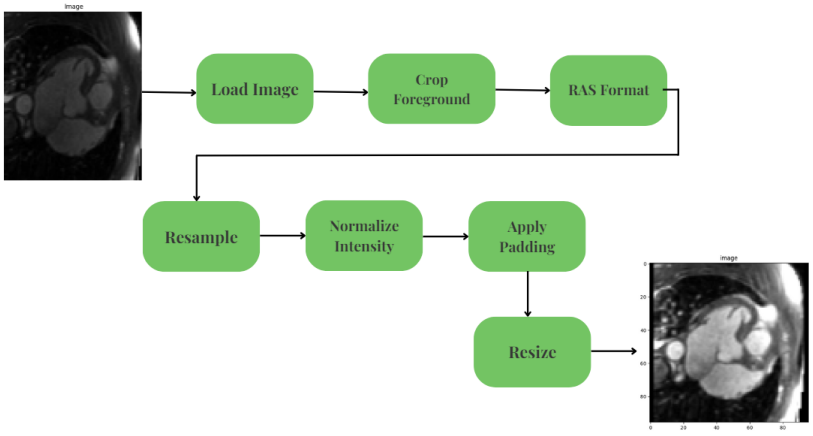

Figure 3.1: *Preprocess Pipeline*

Data Augmentation

Due to the limited training and validation dataset size, data augmentation was essential. Data augmentation aims to increase the size and the variety of training images by generating new samples from existing labeled data. By applying various of MONAI's transformations, like random rotation and adding Gaussian noise to the image and corresponding labels, model performance improved and overfitting was prevented. Random rotation alters the orientation of the images within a specified range, simulating different angles at which medical scans may be captured. Adding Gaussian noise to both the images and the labels introduced small, random variations of imperfections such as sensor noise or slight inconsistencies during scanning. This approach effectively simulated a larger dataset, enhancing the limited number of original CT and MRI images available.

3.3 Model Architecture

This section introduces the architecture of the proposed ViTSegmentation model, which combines transformer-based global context with convolutional inductive biases [1, 15, 48]. The model achieves almost an accurate segmentation of complex anatomical structures in 2D and 3D medical images. The model components are described further down using 3D formulations. The structure of the ViTSegmentation model components is illustrated in Figure 3.2

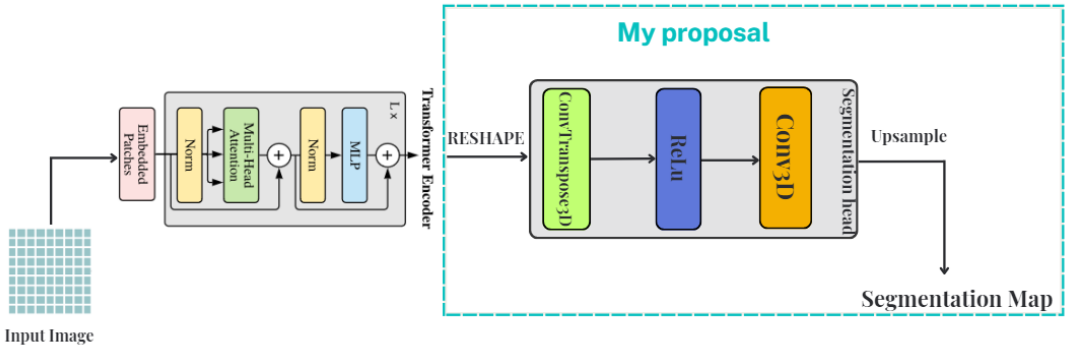

Figure 3.2: *ViTSegment*

3.3.1 Vision Transformer (ViT) Encoder

The ViTSegmentation model's backbone is the Vision Transformer (ViT) encoder from MONAI, is built upon a multi-head self-attention (MHSA) module and fully connected feed-forward network (FFN) [13, 46]. This encoder is the first component that processes the input, transforming it into a form that captures global contextual information. The ViT encoder is made of multiple encoder modules with identical network structure. This is done by stacking multiple encoder modules together. In this illustration there are L encoder layers(L = 12 , ViT-Base Table 3.1 [13]).

Embedded Patches

In the first step, the inputs are transformed into numeric vectors or embeddings [46]. The embeddings allow you to project data into a vector space, where similar data is embedded to similar vectors. Here, the input embeddings are derived from image patches, which effectively transforms an image into a sequence. This allows the data to be processed by a Transformer, that was originally designed for sequence tasks. The block that generates these embeddings it's a fully connected neural network. The initial step involves dividing the input volume $X \in \mathbb{R}^{H \times W \times D \times C}$ into a series of non-overlapping patches,where H,W,D are height width and depth and C number of channels, each of size $P \times P \times P$ [13]. Each patch is flattened into a vector, linearly projected into a high-dimensional space and passed through a fully connected layer to generate token embeddings. This process transforms the image into a sequence of patch embeddings, where each token represents a patch but lacks information about its location within the 3D volume [20]. Mathematically, the number of tokens, N, is given by:

$$N = \frac{H \cdot W \cdot D}{P^3}$$

where H, W and D are the height, width and depth of the volume, respectively, and P is the patch size.

Positional Embeddings

The Vision Transformer also makes use of positional embeddings, the reason for this is that the attention in the Transformer is positioned independent, so positional embeddings make the model understand where each patch is placed in the original image. They have some nice properties, they are learnable vectors and the distance between inputs is consistent for different lengths. These vectors are constructed as a mix of sine and cosine functions [46]. Positional embeddings are added each token right after they are generates and linearly projected into a high-dimensional space.

Normalization (LN) and Skip Connections

The Norm block in the Transformer is based on layer normalization [49]. It normalizes all inputs in a layer for each individual sample. Layer normalization reduces training time and stabilizes the training. Skip connections are used after the Attention and MLP blocks, improving the performance by as much as 4% in recognition tasks, by propagating representations across layers. These connections primarily enhance the flow of information [47].

Multi-Head Self-Attention (MHSA)

Attention Attention mechanisms in artificial neural networks are inspired by the selective focus process that a human has when trying to understand parts of complex information. It enables models to put a priority on different parts of the input data and to highlight the most relevant parts, which not only enhances understanding but also improves performance [21, 31, 46].

Self-Attention Self-attention is used in sequence-based tasks, assigning different weights to each element in the sequence by considering its relationship to all other elements. It performs three different linear projections of input token vectors, producing key, query, and value vectors [21, 31].

- Query (Q): Represents the feature of interest for each token.

- Key (K): Represents the features that may relate to the current query.
- Value (V): Original information or features to be weighted by the attention mechanism.

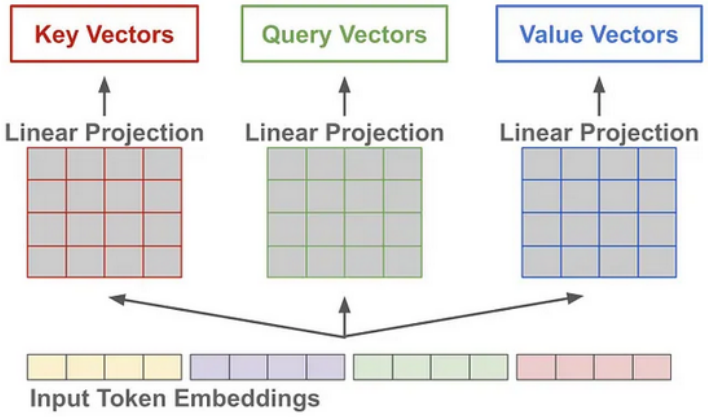

Figure 3.3: *Self-Attention*

Scaled Dot-Product Attention The key component of Multi-Head Self-Attention (MHSA) is the Scaled Dot-Product Attention. The input of the attention mechanism consists the tree token vectors mentioned above (query, key, and value), from which the attention scores are calculated. The scores are calculated by the dot product of the query with all keys. The result is divided by $\sqrt{dk}$, and finally a softmax function is applied to obtain the weights on the values. The scaled dot-product attention used is given by [46]:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V,$$

where d_k is dimension of queries and keys.

Multi-Head Attention Instead of performing a single attention function, there are multiple sets of learnable parameters to produce multiple attention heads, each generating its own set of attention scores (Figure 3.4). These heads are then concatenated and linearly transformed into the expected output. That allows the model to focus on different parts of the input simultaneously [19, 31, 46]. This mechanism enhances the model's ability to:

- Capture complex relationships and patterns in the data, including long-range dependencies.
- Learn different aspects of the input data, as each head focuses on distinct features and patterns.

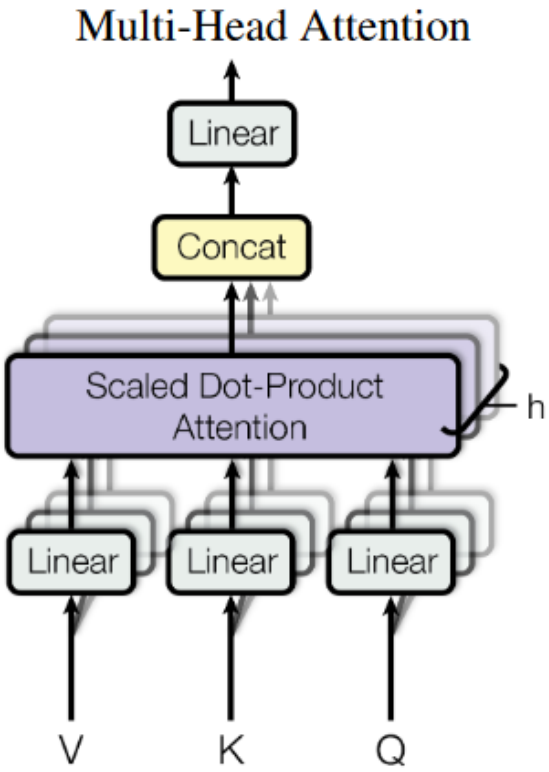

Figure 3.4: *Multi-Head Self-Attention*

Feed-Forward Network (FFN)

In order to increase the model's capacity, each Transformer block includes a type of feed-forward neural network, also known as Multilayer Perceptron (MLP) [28]. This network consists of fully connected neurons with a nonlinear kind of activation function [46]. MLPs consist of an input layer, at least one hidden layer, and an output layer, with each layer applying a weighted sum of inputs followed by an activation function to introduce non-linearity. The MLP consists of two layers, each utilizing a Gaussian Error Linear Unit (GELU) non-linearity.

The Transformer encoder consists of L layers of Multihead Self-Attention (MSA) and Multi-Layer Perceptron (MLP) blocks (Eqs. (3.1), (3.2)). Therefore, the output of the l-th layer can be written as follows [10]:

$$z'_\ell = \text{MHSA}\left(\text{LN}(z_{\ell-1})\right) + z_{\ell-1}, \quad \ell = 1, \ldots, L, \tag{3.1}$$

$$z_\ell = \text{MLP}\left(\text{LN}(z'_\ell)\right) + z'_\ell, \quad \ell = 1, \ldots, L. \tag{3.2}$$

The input embedding is initialized as:

$$z_0 = [x_{\text{class}}, x_1\mathbf{E}, x_2\mathbf{E}, \ldots, x_p\mathbf{E}] + \mathbf{E}_{\text{pos}}, \quad \mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}, \quad \mathbf{E}_{\text{pos}} \in \mathbb{R}^{(N+1) \times D}.$$

Finally, the output representation is computed as:

$$y = \text{LN}(z_L).$$

Here, $\text{LN}(\cdot)$ denotes the layer normalization operator, and z_L is the encoded image representation.

3.3.2 Proposal Decoder (ViTSegment)

After passing through the ViT encoder the output tokens are reshaped to reconstruct the $3D$ feature map. This reshaped structure goes through the segmentation head, which is actually what most Fully Convolutional Networks (FCN) follow in the decoder phase and starts with a $3D$ Transposed Convolution [2, 20, 36, 45].

ConvTransposed3D

The $3D$ Transposed Convolution is used in a lot of models in the decoder phase, it is an up-scaler that increases the input to a higher spatial resolution while reducing the number of channels from $hidden\_size$ to $hidden\_size//2$ (Table 3.1) [14, 18].

$$X_{\text{upsampled}} = \text{ConvTransposed3D}(X_{\text{reshaped}})$$

ReLu

The ReLU activation function is applied to introduce non-linearity into the model and ensure that negative values are set to zero, which helps with better feature extraction and helps the model learn complex patterns.

$$X_{\text{ReLu}} = \text{ReLu}(X_{\text{upsampled}})$$

Conv3D

To obtain the information after a transposed convolution a convolution must be used [14, 36, 45]. The Conv3D projects the feautures from the higher dimensional space that now are to the number of classes for the segmentation (reduces the number of channels from $hidden\_size//2$ to $num\_classes$). This step helps in capturing finer anatomical details.

$$X_{\text{Conv3D}} = \text{Conv3D}(X_{\text{ReLu}})$$

As a final step the output of the segmentation head is upsampled to match the original input resolution and dimension. The Upsample function uses trilinear interpolation (since the input is 3D) to achieve this. In summary, the segmentation head of ViT-Segmentation implements image segmentation by semantically analyzing the features at each location during the decoding process and generating segmentation mappings.

$$\text{Segmentation Map} = \text{Upsample}(X_{\text{Conv3D}})$$

The ViTSegment model combines the mentioned encoder-decoder structure, making it effective for 3D segmentation tasks. The encoder captures global context which is helpful for understanding large structures like organs, while the convolutional decoder enhances local spatial details, capturing structures like tissue boundaries. This approach gives a simple method to use Vision Transformers in medical imaging segmentation tasks other than classification that are usually used.

3.4 Model Training

The Vision Transformer (ViT) encoder that was used in this study was executed based on the ViT-Base architecture. This architecture consists of 12 transformer layers, a hidden size of 768, an MLP size of 3072, and 12 attention heads, which is 86 million trainable parameters. The comparison of different variants of the Vision Transformer model, including ViT-Base, is presented in Table 3.1 [13].

Model	**Layers**	**Hidden size D**	**MLP size**	**Heads**	**Params**
ViT-Base	12	768	3072	12	86M
ViT-Large	24	1024	4096	16	307M
ViT-Huge	32	1280	5120	16	632M

Table 3.1: *Details of Vision Transformer model variants.*

3.4.1 Loss Function

The loss function that was decided to be used for this segmentation task is DiceCELoss from MONAI. This loss function computes both the Dice Loss and Cross-Entropy Loss and returns the weighted sum of these two losses [20]. This method optimizes the segmentation accuracy.

- **Dice Loss** ensures overlap between the predicted and the ground-truth segmentation masks.
- **Cross-Entropy Loss** focuses on the differences of information content between the predicted and the ground-truth segmentation masks.

$$\text{DiceCELoss} = w_0 \frac{2\sum_{c=1}^{N} p_c y_c}{\sum_{c=1}^{N} p_c^2 + \sum_{c=1}^{N} p_c^2 y_c^2} - w_1 \sum_{c=1}^{N} y_c \log(p_c) \qquad [3,\ 12]$$

Key Parameters of DiceCELoss

1. **Weight for Dice Loss (w_0)**: The contribution of Dice Loss to the final loss function. The higher values of w_0 give priority to Dice Loss and highlight the overlap between predictions and ground truth.

2. **Weight for Cross-Entropy Loss (w_1)**: The contribution of Cross-Entropy Loss to the final loss function. Like in (w_0), higher values of w_1 prioritize the Cross-Entropy Loss, which helps ensuring accuracy in predictions.

3. **Predicted Probability (p_c)**: Gives for a certain class c the confidence (probability) of the model regarding its prediction.

4. **Ground Truth (y_c)**: Represents the label of the ground truth class for the class c, typically encoded with one hot.

5. **Number of Classes (N)**: The total number of classes in the dataset.

3.4.2 Optimization Method

As an optimizer, AdamW Optimizer was chosen, an enhanced version of the traditional Adam Optimizer [38]. AdamW separates the weight decay term from the gradient update, allowing the learning rate and regularization to be optimized independently. This decoupling results to more stable and faster convergence during training, and it is especially beneficial for models that are trained on medical imaging datasets, where stability is essential. For this training, the following hyperparameters were chosen based in prior experimentation:

- Learning Rate (α): 1×10^{-4}, selected to ensure stable convergence.
- Weight Decay (λ): 0.01, for effective regularization without stalling the model's learning.
- Momentum Parameters (β_1, β_2): Default values of 0.9 and 0.999, consistent with Adam and AdamW implementations.

- AMSGrad: set to `True`, ensuring that the optimizer uses the AMSGrad variant and keeps a history of past gradients.

3.4.3 Learning Rate Scheduler

A learning rate scheduler was implemented to adjust the learning rate during the training [22, 34]. This scheduler combines a linear warm-up phase followed by an exponential decay, ensuring that the model has enough time to adapt initially and gradually reduces the learning rate as training progresses [35, 39]. The learning rate scheduler works as follows:

- **Warm-up Phase**: During the first 5 epochs (defined by the `warmup_epoch` parameter), the learning rate increases linearly from 0 to the initial learning rate (`init_lr` $= 1 \times 10^{-4}$) to allow the model to start training more gradually, avoiding large updates in the early stages.
- **Exponential Decay**: After the warm-up phase, the learning rate decays exponentially with a factor of 0.1 to ensure the optimizer performs finer updates. This phase continues until the maximum number of epochs is reached (defined by the `max_epoch` parameter).

The learning rate at each epoch e is computed as:

$$\text{lr} = \begin{cases} \frac{\text{init\_lr} \cdot (e+1)}{\text{warmup\_epoch}} & \text{if } e < \text{warmup\_epoch} \\ \text{init\_lr} \cdot 0.1^{\frac{(e - \text{warmup\_epoch})}{\text{max\_epoch} - \text{warmup\_epoch}}} & \text{if } e \geq \text{warmup\_epoch} \end{cases}$$

This scheduler is applied at each step, and the learning rate is updated in the optimizer.

3.4.4 Cross-Validation and Data Splitting

Separate models were trained for CT and MRI datasets to ensure optimal performance for each imaging modality, given their distinct characteristics, such as resolution and

noise levels. In order to assess the model's performance , 5-fold cross-validation was employed for the training and validation phases. This ensures that the model's performance is evaluated across different random subsets of the data, providing more reliable results compared to a single train-test split. The process was implemented using MONAI's `CacheDataset` and cross-validation splits [5].

Dataset Splitting

The dataset was first randomly split into five folds. Each fold consists of a subset of the data that will be used as a validation set in one iteration, while the remaining four folds are used for training. This procedure is repeated five times, such that each fold acts as the validation set once. The `CVDataset` class was implemented as a base class to generate the datasets for cross-validation. It extends the `CacheDataset` class and ensures that the dataset is split accordingly [5].

Cross-Validation Implementation

The `CrossValidation` class from MONAI was used to handle the cross-validation process. For each fold, the training and validation datasets are selected, augmented, and fed into the model during training. For each fold:

- The dataset is split into training and validation sets.
- The training set is further augmented.
- The augmented training set is concatenated with the original training data to form the final training dataset.
- Separate DataLoaders are created for both training and validation sets, with a batch size of 1 for inference.

Post-Processing Transforms

After each prediction, post-processing transforms are applied to convert the output into the desired format for evaluation. These transforms include applying activation

functions and converting the output to one-hot encoding.

Test Dataset

Once the model is trained, it is tested on a separate test dataset. This dataset consists of 1 left-out image with ground-truth label and 40 test images with no label.

3.4.5 Training Details

The model was trained for 25000 iterations with a batch size of 1, due to memory restrictions and it took about 5 hours (because of 5-fold cross validation so 1 hour per training). All experiments were performed on an Intel Core i7-13700K processor, featuring 16 cores and 24 threads, coupled with an AMD Radeon RX 7900 XTX GPU with 24 GB of VRAM. To train on the AMD GPU in Windows 11, the recommended approach is using Windows Subsystem for Linux (WSL) along with ROCm (Radeon Open Compute). This set up enables compatibility with PyTorch and allows access to AMD's GPU acceleration.

Chapter 4

Evaluation and Experimental Results

This chapter presents the evaluation metrics and results of the implemented ViTSegment model. The experiments are designed to validate the model's performance on the chosen datasets and tasks, with comparisons made against baseline Convolutional Neural Networks (CNNs) and other state-of-the-art methods.

4.1 Evaluation Metrics

When performing a segmentation with a model architecture, it is essential to define metrics to evaluate the performance of the model. The evaluation is carried out using the MM-WHS dataset (see Chapter 3 section 3.2.1), consisting of MRI and CT volumes. Various evaluation metrics are employed to evaluate the model using labeled data.

4.1.1 Confusion Matrix

Confusion Matrix is a visualization method that measures the performance of a classification model. In image segmentation, the confusion matrix does the same as in classification tasks by measuring how well the model predictions match the ground truth [16, 26]. It is used, mostly, to calculate other model evaluating metrics, such as recall and precision. In multi-class tasks it is a $n \times n$ matrix , where n is the number of classes, so for a binary classifier it may look like this :

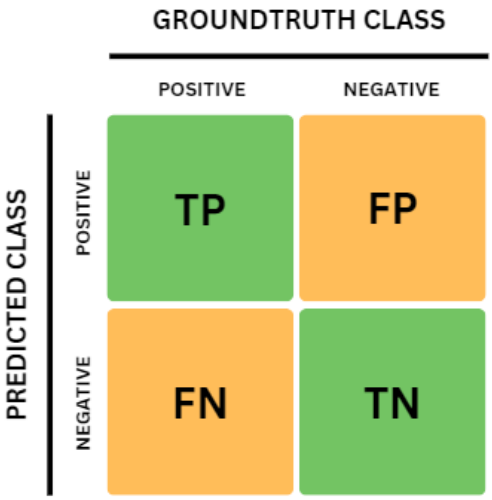

Figure 4.1: *Confusion Matrix*

Where,

- **TP**: The model correctly predicted a positive class when it is indeed positive.
- **TN**: The model correctly predicted a negative class when it is indeed negative.
- **FP**: The model incorrectly predicted a positive class when it should have been negative.
- **FN**: The model incorrectly predicted a negative class when it should have been positive.

The confusion matrix provides the foundation for calculating evaluation metrics, such as:

Accuracy

Accuracy is the proportion of correct predictions to the total number of predictions. Although it is very informative on the performance of the model, it can be misleading

for imbalanced classes [16, 23, 26].

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision

Precision score is the number of true positive predictions among all positive predictions or else, it is the ratio of positive class predictions that actually belong to the target class. It measures the accuracy of positive predictions [16, 23, 26].

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall

Recall, also known as sensitivity, it is the proportion of positive predictions for target class to the actual number of positive instances. A high recall means the model successfully identifies most of true positive instances [16, 23, 26].

$$\text{Recall} = \frac{TP}{TP + FN}$$

4.1.2 Dice Similarity Coefficient (F1 Score)

While the above mentioned metrics provide a satisfying measurement of the model's performance, in segmentation a more specific metric is Dice Similarity Coefficient. It is a widely used metric due to its simplicity and effectiveness. The DSC is a measure of the similarity between the predicted segmentation mask and the ground truth mask. It takes values between 0 and 1, with 1 indicating perfect overlap and 0 no overlap [32]. It is defined as:

$$DSC = \frac{2 \times AreaOfOverlap}{TotalArea} = \frac{2 \times}{\text{prediction} + \text{groundtruth}}$$

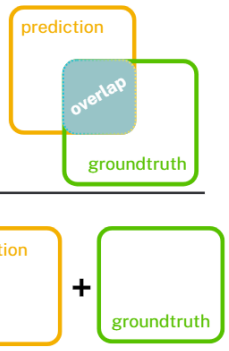

Figure 4.2: *Dice Similarity Coefficient [23]*

In this case of multi-class segmentation, the DSC is calculated separately for each class (e.g., Left Ventricle, Right Ventricle, etc.) and then averaged to provide a general performance score, as do all the metrics.

4.1.3 Intersection over Union (IoU)

The Intersection-over-Union (IoU), also known as Jaccard index or Jaccard similarity coefficient is also a common metric used to evaluate the performance of the image segmentation model. It measures the amount of the intersection of the predicted segmentation mask and the ground truth mask to their combined areas. In general, IoU values closer to 1 indicate greater accuracy in segmentation [42].

$$IoU = \frac{OverlapingVolume}{VolumeOfUnion} = \frac{}{\text{prediction} \quad \text{groundtruth}}$$

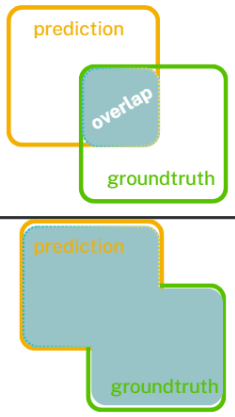

Figure 4.3: *Intersection over Union (IoU) [23]*

4.1.4 Hausdorff Distance (HD)

Another critical metric is the Hausdorff Distance (HD), which it measures the greatest distance from any point on the boundary of the groundtruth segmentation to the closest point on the boundary of the predicted segmentation. The process is repeated for both directions, and the maximum of the two distances is taken (worst-case scenario). The model has to not only detect the structures but also correctly delineates their boundaries [6]. As shown in Figure 4.4, for two sets of points X and Y, the ***one-sided Hausdorff distance*** from X to Y is given by:

$$hd(X, Y) = \max_{x \in X} \min_{y \in Y} \|x - y\|_2$$

Here is used the Euclidean distance between two points, but any distance can be used. The one-sided HD quantifies how far X is from Y, but is not commutative. The ***total Hausdorff distance*** or else the bidirectional is given by:

$$HD(X, Y) = \max\{hd(X, Y), hd(Y, X)\}$$

A small HD indicates that the boundaries of the predicted segmentation are closer to the groundtruth boundaries.

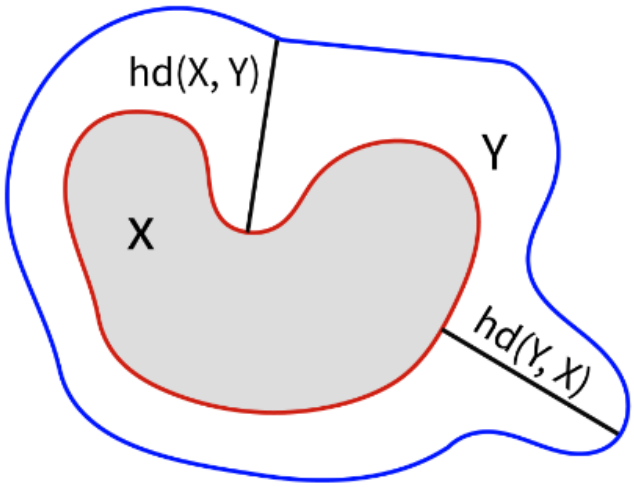

Figure 4.4: *Illustration of the Hausdorff distance between two sets of points X and Y. [6]*

4.2 Results

The ViTSegment model is evaluated on both the CT and MRI datasets. Below are presented the result for each modality separately. First, to observe the training process of the model two key metrics were tracked across iterations:
Iteration Average Loss: It reflects the ability of the model to decrease the variance between the groundtruth and predicted segmentation masks.
Validation Mean Dice: It evaluates the model's performance on validation (unseen) data during the training process.
Then the evaluation metrics mentioned earlier (HD,IoU, etc.) were applied to measure the performance. They were calculated for each class and then averaged to summarize the overall performance without taking the background class into account. At the end, visual results are shown.

4.2.1 CT Dataset

In this subsection the CT dataset results are introduced and discussed.

Loss-Dice Diagrams

Figure 4.5 illustrates both the training loss and the validation Dice score.

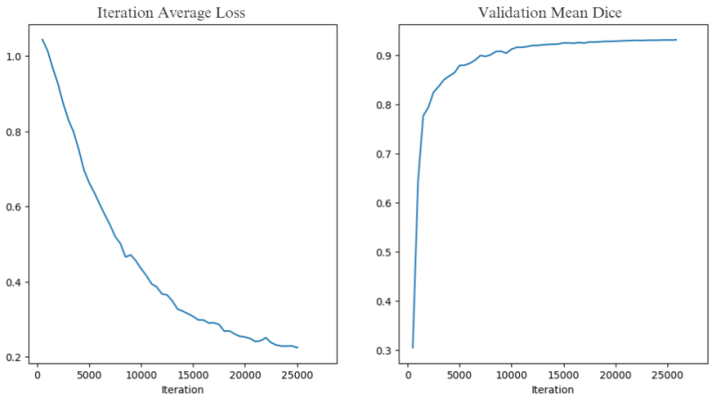

Figure 4.5: *The training loss and validation Dice score (CT)*

The loss decreases with the training iterations and shows that the model is making the best possible use of the DiceCELoss function. The model is learning and reducing as steadily as possible the variance in the loss between the predicted and groundtruth masks. Eventually, the loss stabilizes at the end. On the other hand, the mean dice increases and towards the end of the training is being stabilized also. This shows that the model is improving over the iterations on validation data.

Metrics

The DSC , IoU and HD metrics are computed for each heart substructure and summarized together for the average value, as shown in Table 4.1.

Heart Substructure	Dice Similarity Coefficient (%)	IoU (%)	Hausdorff Distance (mm)
Left Ventricle (LV)	93.66 ± 1.60	87.67 ± 2.70	2.31 ± 0.34
Right Ventricle (RV)	94.56 ± 0.57	87.95 ± 2.11	2.92 ± 1.19
Left Atrium (LA)	94.58 ± 1.90	90.34 ± 1.96	3.03 ± 1.01
Right Atrium (RA)	93.48 ± 0.80	86.95 ± 1.31	3.48 ± 0.93
Myocardium (MYO)	91.40 ± 1.89	85.59 ± 2.02	2.84 ± 0.81
Ascending Aorta (AO)	92.90 ± 1.50	86.81 ± 1.11	2.71 ± 0.38
Pulmonary Artery (PA)	87.93 ± 1.80	76.67 ± 4.88	6.74 ± 2.24
Average	**92.65 ± 2.17**	**86.41 ± 4.50**	**3.43 ± 1.80**

Table 4.1: *Segmentation performance for each heart substructure in validation, on the CT dataset.*

Each substructure and the average of them are achieving a high DSC of 92.65 ± 2.17%, that indicates the effectiveness of the model. All substructures' values exceed 90%, except the Pulmonary Artery with a dice of 87.93 ± 1.80%. The IoU metric looks similar with DSC, with an average IoU of 86.41 ± 4.50%. While the IoU scores for most substructures remain close to or above 85%, the Pulmonary Artery exhibits the lowest IoU at 76.67± 4.88%. Last but not least, the HD metric gives the model's boundary delineation, with an average of 3.43± 1.80mm. However again the Pulmonary Artery displays the greatest distance with a score of 6.74 ± 2.24mm, indicating challenges in precisely delineating this structure's boundaries. In total, the model seems to perform quite well, also taking into account the rest of evaluation metrics in Table 4.2

Metric	Average Value (%)
Accuracy	99.5 ± 0.08
Precision	92.43 ± 3.05
Recall (Sensitivity)	92.88 ± 2.39

Table 4.2: *Performance metrics for the model in validation, on CT dataset.*

Visual Results

During the training process, one random image (ct_train_1001) was left out from the training dataset so it can be used as a test image. This approach was taken because the provided test images did not include a segmentation groundtruth. The groundtruth segmentation for this image was compared to the predicted (see Figure 4.6), and the Dice score was computed to quantify the model's accuracy (see Table 4.3).

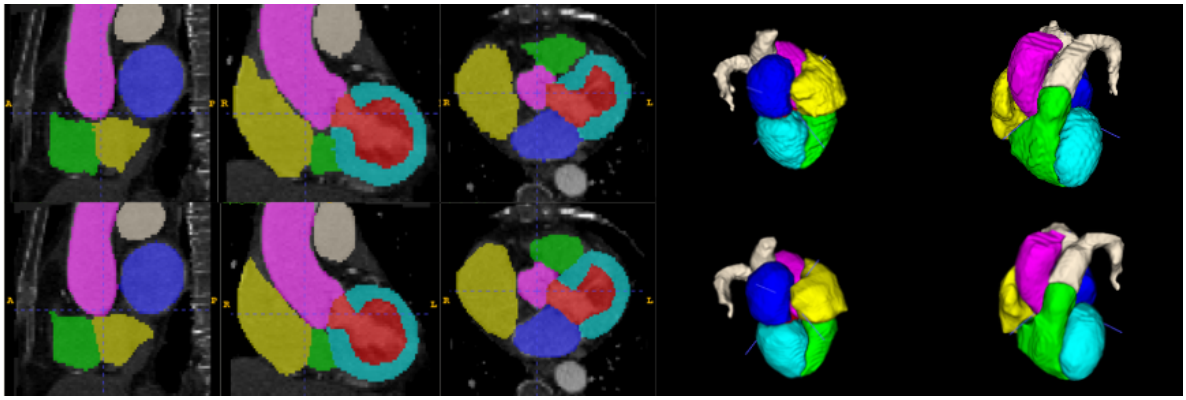

Figure 4.6: *Results of unseen labeled CT Image (ct_train_1001). Groundtruth segmentation (top row) compared with the predicted segmentation (bottom row).*

Heart Substructure	Dice Similarity Coefficient (%)
Left Ventricle (LV)	86.50
Right Ventricle (RV)	87.50
Left Atrium (LA)	94.73
Right Atrium (RA)	90.10
Myocardium (MYO)	86.71
Aorta (AO)	91.72
Pulmonary Artery (PA)	84.20
Average	**90.10**

Table 4.3: *DSC score for CT Image: ct_train_1001 in testing.*

The model when tested it has a lower value of DSC compared to the validation results. However, the differences are not significant and this shows that the model exhibits well overall. In Figure 4.7 it is shown the predicted segmentation from the test set that do not have a groundtruth segmentation.

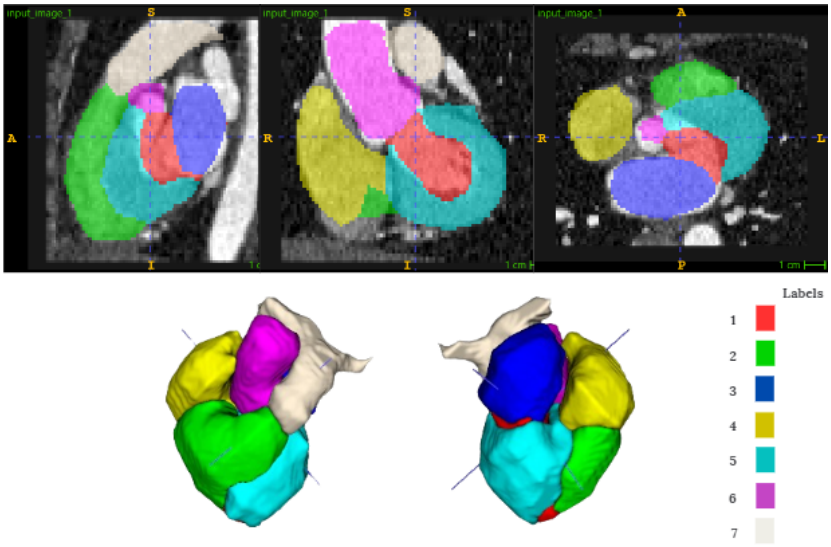

Figure 4.7: *Predicted segmentation for ct_test_2002 image (image from test set without groundtruth segmentation).*

The results visually suggest that the model successfully identifies the primary structures of the heart, including the ventricles and atria. However, subtle inaccuracies exist. For example, the substructure boundaries are not always precise, sometimes they extend slightly beyond or fall short of the actual boundaries.

4.2.2 MRI Dataset

For the MRI dataset, the challenge was bigger because of the lower resolution. The results in MRI are following the same concept as in CT dataset.

Loss-Dice Diagrams

The Figure 4.8 presents the training loss and validation Dice score for the MRI dataset.

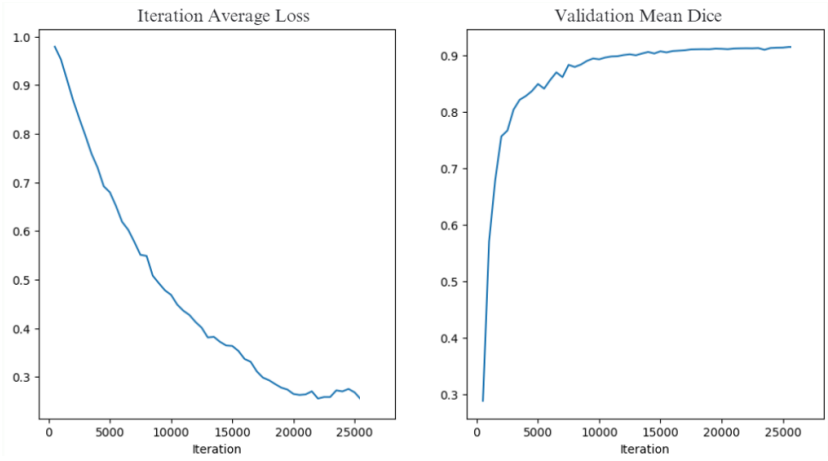

Figure 4.8: *The training loss and validation Dice score (MRI)*

The training loss shows a steady decline over iterations, indicating effective learning. Meanwhile, the validation Dice score increases consistently, stabilizing towards the later iterations. This demonstrates the model's ability to improve on validation data during training.

Metrics

The DSC , IoU and HD metrics are computed for each heart substructure and summarized together for the average value, as shown in Table 4.4.

Heart Substructure	Dice Similarity Coefficient (DSC %)	IoU %	Hausdorff Distance (mm)
Left Ventricle (LV)	93.26 ± 1.65	89.40 ± 1.55	2.91 ± 0.48
Right Ventricle (RV)	91.95 ± 2.06	86.71 ± 2.63	5.52 ± 4.55
Left Atrium (LA)	92.66 ± 1.63	85.94 ± 2.29	3.05 ± 0.41
Right Atrium (RA)	92.91 ± 2.14	84.48 ± 2.28	4.01 ± 1.37
Myocardium (MYO)	90.27 ± 2.05	83.20 ± 1.36	5.53 ± 1.63
Ascending Aorta (AO)	91.45 ± 0.91	84.49 ± 0.99	3.03 ± 0.17
Pulmonary Artery (PA)	87.97 ± 1.76	77.34 ± 5.08	5.30 ± 1.76
Average	**91.50 ± 1.72**	**84.51 ± 3.46**	**4.19 ± 1.14**

Table 4.4: *Segmentation performance for each heart substructure in validation, on the MRI dataset.*

As it's shown, the model exhibits a DSC of 91.50±1.72%, a little less than the CT dataset. The Pulmonary Artery has also the lower value of the rest substructures (87.97±1.76) as in CT. For the IoU score, the model reached 84.51±3.46% which is also lower than the CT's IoU, with the Pulmonary Artery again scoring the lowest value (77.34±5.08%). Finally the HD metric of MRI dataset is 4.19±1.14, a less promising value compared to CT. The difference here is that more than one substructure exhibits considerably poor HD values, reflecting the additional challenges presented by MRI data.

Metric	Average Value (%)
Accuracy	99.3 ± 0.09
Precision	90.95 ± 3.05
Recall (Sensitivity)	92.07 ± 1.99

Table 4.5: *Performance metrics for the model in validation, on MRI dataset.*

Visual Results

During the training process, one random image (mri_train_1017) was left out from the training dataset so it can be used as a test image. This was done because the provided test images (mri test set) did not include a segmentation groundtruth. The

groundtruth segmentation for this image was compared to the predicted (see Figure 4.9), and the Dice score was computed to quantify the model's accuracy for that image (see Table 4.6).

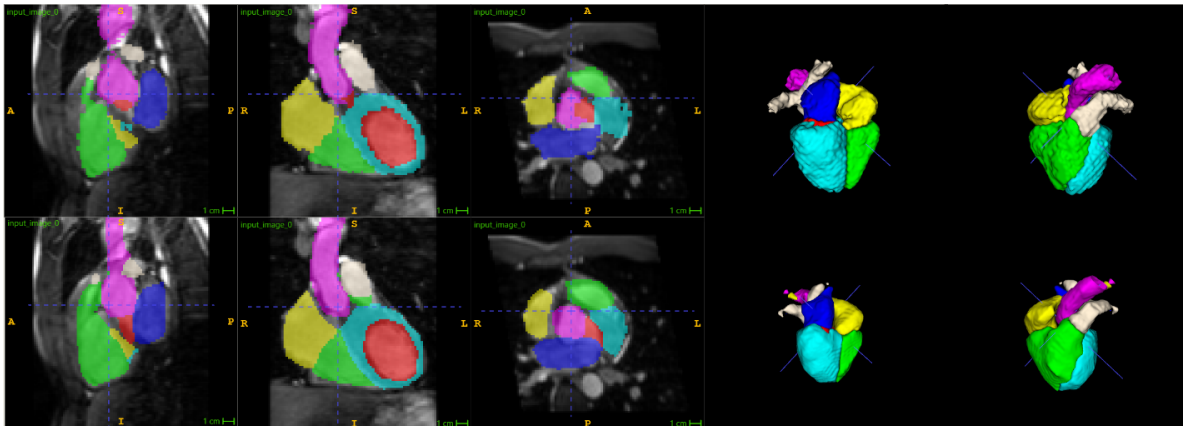

Figure 4.9: *Results of unseen labeled MRI Image (mri_ train_ 1017). Groundtruth segmentation (top row) compared with the predicted segmentation (bottom row).*

Heart Substructure	Dice Similarity Coefficient (%)
Left Ventricle (LV)	90.30
Right Ventricle (RV)	89.75
Left Atrium (LA)	88.68
Right Atrium (RA)	86.70
Myocardium (MYO)	81.95
Aorta (AO)	80.96
Pulmonary Artery (PA)	75.80
Average	**84.88**

Table 4.6: *DSC score for MRI Image: mri_ train_ 1017 in testing.*

The testing DSC is significantly lower than the validation DSC, this indicates the challenges of this dataset. The MRI dataset needed larger dataset due to noise and lower resolution. Despite this, the predicted segmentation of the image is considered quite good for this dataset. In Figure 4.10 it is shown the predicted segmentation from the test set that do not have a groundtruth segmentation.

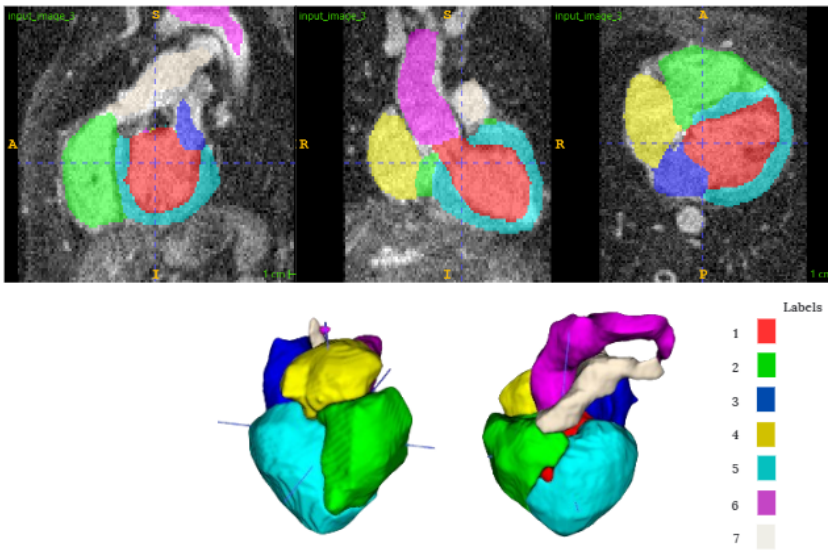

Figure 4.10: *Predicted segmentation for mri_test_2004 image (image from test set without groundtruth segmentation).*

The predicted segmentation in this MRI test image looks overall good with some errors. For example, here also the boundaries are not accurate, and there are issues with the segmentation of the Pulmonary region and the Aorta, which are not clearly delineated, leading to inaccuracies in the results.

4.3 Comparison with Other Architectures

In this section, we compare the performance of the ViTSegment model with two other widely-used segmentation architectures in medical imaging: U-Net and UNETR. All models were trained under identical conditions, using the same dataset, preprocessing, and hyperparameters.

4.3.1 U-Net

The U-Net is a fully convolutional network designed for semantic segmentation. It is one of the most used architectures for medical image segmentation due to its efficiency. It follows the encoder-decoder method, consisting of a contracting path (encoder) and

an expansive path (decoder). This architecture is named after its U-shaped design. The U-Net model was trained under the same conditions as ViTSegment did, it took approximately 10 hours to complete the training and was evaluated the same way as ViTSegment. It was evaluated on both CT and MRI dataset, on CT dataset it exhibited a Dice score of 82.67 ± 8.70% and on MRI 81.30 ± 5.47% (see Table 4.11 and 4.12). Then the model was tested on the same left out image on CT (ct_train_1001) (see Table 4.7) and on MRI (mri_train_1017) (see Table 4.8).

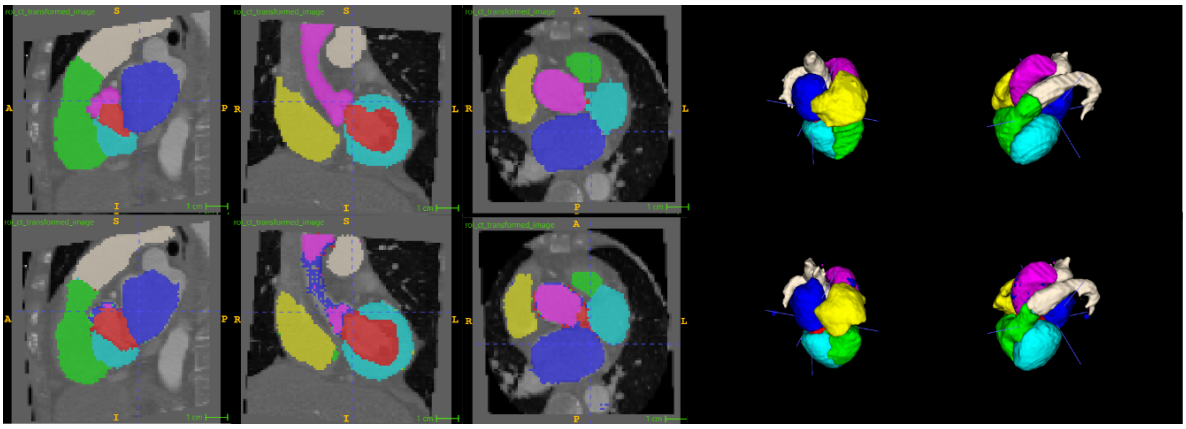

Figure 4.11: *U-Net results of unseen labeled CT Image (ct_train_1001). Groundtruth segmentation (top row) compared with the predicted segmentation (bottom row).*

Heart Substructure	**Dice Similarity Coefficient (%)**
Left Ventricle (LV)	80.80
Right Ventricle (RV)	73.29
Left Atrium (LA)	83.47
Right Atrium (RA)	77.44
Myocardium (MYO)	79.79
Aorta (AO)	86.05
Pulmonary Artery (PA)	77.42
Average	**79.75**

Table 4.7: *DSC score for CT Image: ct_train_1001 in testing, using U-Net.*

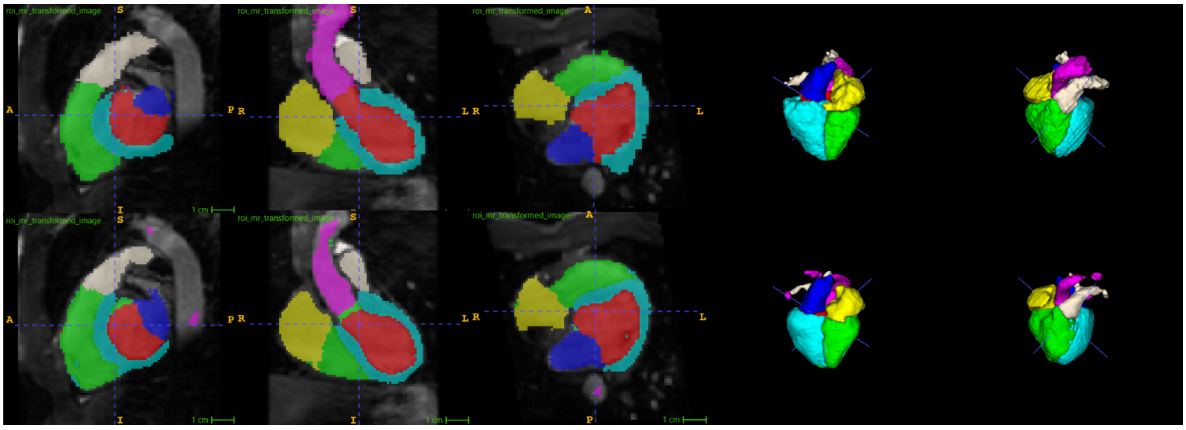

Figure 4.12: *U-Net results of unseen labeled MRI Image (mri_train_1017). Groundtruth segmentation (top row) compared with the predicted segmentation (bottom row).*

Heart Substructure	Dice Similarity Coefficient (%)
Left Ventricle (LV)	89.00
Right Ventricle (RV)	88.04
Left Atrium (LA)	80.48
Right Atrium (RA)	79.61
Myocardium (MYO)	79.32
Aorta (AO)	63.24
Pulmonary Artery (PA)	62.66
Average	**77.48**

Table 4.8: *DSC score for MRI Image: mri_train_1017 in testing, using U-Net.*

4.3.2 UNETR

The UNETR model is a hybrid of U-Net model with transformer encoders. It can be seen as a more complex variant of ViTSegment (proposal), which also integrates vision transformers for encoders. The training for UNETR was also under the same conditions as in the U-Net and ViTSegment. The overall Dice score of the model on CT dataset was 86.33 ± 0.74% and on MRI 84.94 ± 6.25% (see Table 4.11 and 4.12). The model was also tested using the same left out image on the CT dataset ((ct_train_1001, see Table 4.13) and the MRI dataset (mri_train_1017, see Table 4.14).

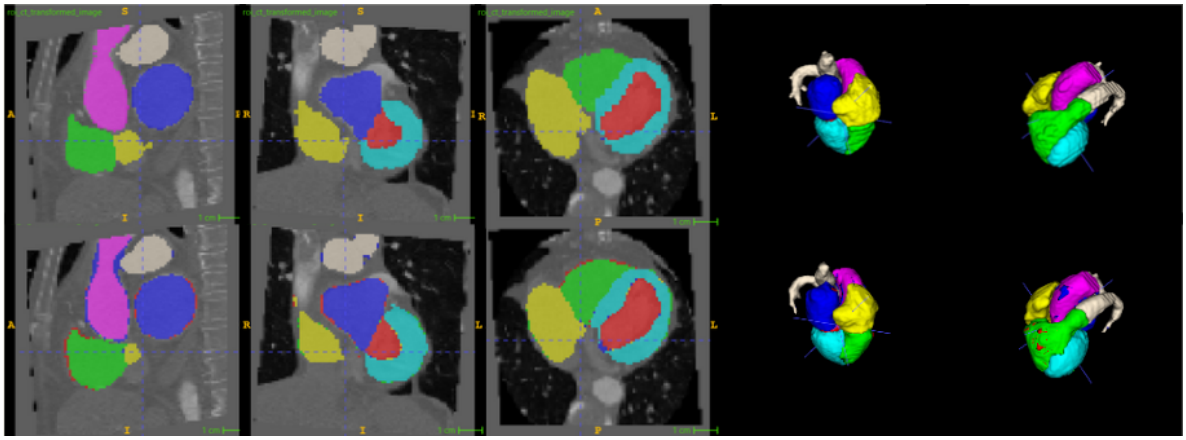

Figure 4.13: *UNETR results of unseen labeled CT Image (ct_train_1001). Groundtruth segmentation (top row) compared with the predicted segmentation (bottom row).*

Heart Substructure	**Dice Similarity Coefficient (%)**
Left Ventricle (LV)	78.77
Right Ventricle (RV)	83.35
Left Atrium (LA)	87.71
Right Atrium (RA)	83.82
Myocardium (MYO)	86.34
Aorta (AO)	94.06
Pulmonary Artery (PA)	87.78
Average	**87.61**

Table 4.9: *DSC score for CT Image: ct_train_1001 in testing, using UNETR.*

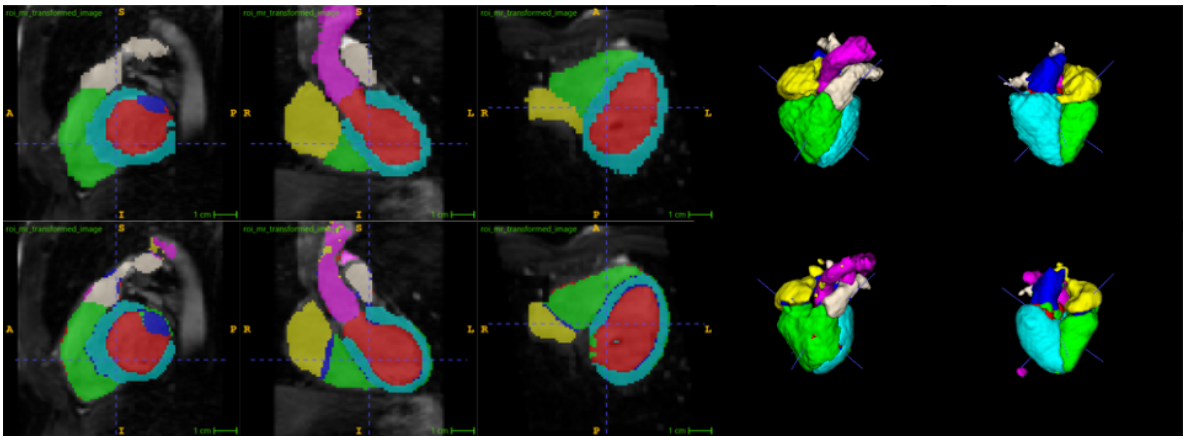

Figure 4.14: *UNETR results of unseen labeled CT Image (mri_train_1017). Groundtruth segmentation (top row) compared with the predicted segmentation (bottom row).*

Heart Substructure	Dice Similarity Coefficient (%)
Left Ventricle (LV)	80.20
Right Ventricle (RV)	78.43
Left Atrium (LA)	76.38
Right Atrium (RA)	85.81
Myocardium (MYO)	79.83
Aorta (AO)	81.14
Pulmonary Artery (PA)	79.97
Average	**80.14**

Table 4.10: *DSC score for MRI Image: mri_train_1017 in testing, using UNETR.*

4.3.3 Computational Efficiency and Time

The UNETR, as a more complex model, took approximately 24 hours to complete. The U-Net model although a light-weighted model it took about 10 hours for the training process. In terms of training time and model complexity, the ViTSegment was the fastest (completed in 5 hours) and less complex model among the three architectures. The reasons that the ViTSegment outcompeted the U-Net are :

- The U-Net operates on the entire image while ViTSegment on flatten image patches.
- The U-Net process the input sequential while ViTSegment in parallel, making it faster.

Comparing the Dice scores (see Table 4.11 and 4.12) the ViTSegment demonstrated the best performance with a Dice score of 92.65 $\pm$ 2.17% on the CT dataset and 91.50 $\pm$ 1.72% on the MRI. Although the UNETR performed quite well on some heart substructures, it required more training time because of its architecture complexity. The CT dataset, across all three models, performed better than the MRI, which also happens in all the related previews works. This is often attributed to the small dataset and the low resolution of the images. Additionally, it is noticed that the Pulmonary artery and the Ascending aorta perform poorly in the test images and that may be due to ambiguities for the definition of them, which are hard to consistently segment manually (leading to inter-observer errors)a challenge also noted in previous works done.

Dice Similarity Coefficient (%) on CT dataset

Heart Substructure	**U-Net**	**UNETR**	**ViTSegment**
Left Ventricle (LV)	87.65 ± 5.73	82.20 ± 4.99	93.66 ± 1.60
Right Ventricle (RV)	85.60 ± 1.83	88.32 ± 1.14	94.56 ± 0.57
Left Atrium (LA)	82.55 ± 3.21	71.18 ± 5.30	94.58 ± 1.90
Right Atrium (RA)	84.15 ± 5.41	85.82 ± 5.36	93.48 ± 0.80
Myocardium (MYO)	84.29 ± 4.95	87.75 ± 3.68	91.40 ± 1.89
Ascending Aorta (AO)	80.16 ± 1.82	95.02 ± 2.94	92.90 ± 1.50
Pulmonary Artery (PA)	74.33 ± 7.21	94.06 ± 1.16	87.93 ± 1.80
Average	82.67 ± 8.70	86.33 ± 0.74	**92.65 ± 2.17**

Table 4.11: *DSC results in validation, on CT dataset for various heart structures using U-Net, UNETR, and ViTSegment models.*

Dice Similarity Coefficient (%) on MRI dataset

Heart Substructure	**U-Net**	**UNETR**	**ViTSegment**
Left Ventricle (LV)	80.97 ± 3.54	81.90 ± 4.06	93.26 ± 1.65
Right Ventricle (RV)	78.38 ± 5.92	80.98 ± 5.65	91.95 ± 2.06
Left Atrium (LA)	70.31 ± 6.69	73.36 ± 5.59	92.66 ± 1.63
Right Atrium (RA)	87.77 ± 3.57	89.82 ± 3.92	92.91 ± 2.14
Myocardium (MYO)	80.48 ± 7.21	85.21 ± 4.27	90.27 ± 2.05
Ascending Aorta (AO)	86.18 ± 5.01	92.12 ± 0.94	91.45 ± 0.91
Pulmonary Artery (PA)	85.03 ± 2.30	91.17 ± 0.82	87.97 ± 1.76
Average	81.30 ± 5.47	84.94 ± 6.25	**91.50 ± 1.72**

Table 4.12: *DSC results in validation, on MRI dataset for various heart structures using U-Net, UNETR, and ViTsegment models.*

Chapter 5

Discussion

In this thesis, a fully automatic multi-label segmentation for CT and MRI data is presented, using ViTSegment, the proposed method of this study. ViTSegment is a Vision Transformer-based hybrid model designed for whole heart image segmentation. The model integrated a Vision Transformer-based encoder with a global receptive field from the very first layer allowing them to capture long-range dependencies and a convolutional decoder with local spatial feature to enhance the boundary detection. In the decoder phase, transposed convolution was employed for upsampling, ReLu activation for feature extraction, and convolution layer for further refinement of the feature map. The training process involved a combination of Dice and Cross-Entropy loss function (DiceCELoss), with hyperparameter tuning to avoid overfitting, and utilized a learning rate scheduler for stabilization. The data set consists of 20 labeled CT and 20 labeled MRI heart volumes from the MICCAI MM-WHS challenge, with data augmentation techniques used to address the limitation of the dataset's size.

The segmentation accuracy of the model was evaluated using metrics such as the Dice similarity coefficient (DSC), Intersection over Union (IoU), and Hausdorff Distance (HD). Some of the key results were the DSC of 92.65 $\pm$ 2.17% on CT dataset and 91.50 $\pm$ 1.72% on MRI dataset. The predicted segmentation map was accurate in the identification of some heart substructures, such as ventricles and atria. The segmentation for smaller substructures, such as the pulmonary artery, presented some challenges achieving DSC of 87.93 $\pm$ 1.80% on CT and 87.97 $\pm$ 1.76% on MRI dataset. The pulmonary artery was the challenging class, likely due to difficulties in class imbalance

and inter-observer variability.

In addition to validation, the model's performance was tested on unseen images from both CT and MRI datasets. One randomly selected labeled image was left out of the training process for both datasets to measure the DSC, while the model performance was also visually evaluated on the test dataset, which contained images without segmentations. To be more precise, the ct_train_1001 image was excluded from the CT dataset, and the mri_train_1017 image from the MRI. On the testing CT image the average DSC was 90.10% a little lower than the validation average (92.65%) while the pulmonary artery once again exhibited the lowest dice score. These results are represented also in the comparison image (4.6), where the overall segmentation matches quite well with the ground truth, but the pulmonary class exceeds a little from the ground truth boundaries, and it lacks the finishing details, leading to reduced accuracy in this region.

For the MRI testing image results showed similar findings. The model exhibited an average DSC of 84.88%, which was also lower than the validation average (91.50%) and a bad performance on the pulmonary with only 75.80% of dice score. In the comparison image (4.9), several inaccuracies are observed, including incomplete segmentation of the pulmonary structure and overlap between the left atrium and the aorta.

Based on the visual analysis of the images from the test dataset of both CT and MRI (Figures 4.7 and 4.10), the predicted segmentations demonstrate strong accuracy for the given images with anatomical realism and clear boundaries delineations between different regions. Although, some potential concerns might be the slight mislabelling or over-smoothing, which affects the accuracy of smaller details.

Despite all the above, ViTSegment outperformed traditional models such as U-Net and UNETR on both CT and MRI datasets in validation and test results. With U-Net achieving validation Dice scores of 82.67% on CT and 81.30% on MRI, and UNETR achieving 86.33% and 84.94% respectively. These two models were tested only with the training-left-out labeled images, where they also underperformed compared to ViTSegment (see Tables (4.7,4.8,4.9,4.10)).

Tables 5.1 and 5.2 present a comparison of the proposed method, ViTSegment, against results from other studies employing various models. It is observed that all the models

performed better on CT than on the MRI dataset, which highlights the challenges associated with the MRI dataset, as mentioned in this thesis earlier. Furthermore, certain substructures consistently exhibit high DSC scores, which indicates that these regions may be easier to segment. It is also notable that the pulmonary artery consistently performs worse than all other substructures across nearly all models.

This pattern is observed both with the proposed model,ViTSegment, and with the two other models, U-Net and UNETR, used for comparison in Chapter 4 of this thesis. In conclusion, ViTSegment outperforms other models in most substructures on the CT dataset and across all substructures on the MRI dataset, also achieving the highest average DSC on both datasets.

Dice Similarity Coefficient (%) in Validation

Participant	MYO	LA	LV	RA	RV	AO	PA	AVG
SEU (MAS)	87.24	**95.94**	**96.01**	88.36	**93.44**	92.95	79.66	90.51
KTH (U-net & SC)	87.9	90.8	93.5	88.1	82.5	**95.9**	81.5	88.68
CUHK2 (TL+DS+H)	81.86	84.54	87.76	81.53	77.80	94.12	82.62	84.32
UT (E1)	88.2	88.8	88.2	89.0	88.1	82.0	**92.3**	88.09
UCF (MO-MP-CNN)	89.8	92.5	93.0	87.7	88.8	90.9	85.1	89.7
GUT (Seg-CNN)	92.4	87.2	87.9	92.4	87.8	91.1	83.3	88.9
CUHK1 (TL+DS+mDSC)	68.97	90.00	83.42	84.36	62.50	91.51	80.84	80.22
ViTSegment (Our Proposal)	**93.66**	94.56	94.58	**93.48**	91.40	92.90	87.93	**92.65**

Table 5.1: *Dice Similarity Coefficient (DSC) on CT dataset from other studies on the MICCAI challenge. Participants: University or research team (proposed method) [40]*

Dice Similarity Coefficient (%) in Validation

Participant	MYO	LA	LV	RA	RV	AO	PA	AVG
UOL (MAS)	78.1	88.6	91.8	87.3	87.1	87.8	80.4	86.0
KTH (U-net & Shape Context)	80.7	84.7	89.5	82.1	79.5	67.9	74.3	80.8
CUHK2 (TL+DS+hybrid)	74.17	78.66	85.83	81.99	81.91	72.60	69.83	77.86
UT (E1)	88.4	82.4	75.5	85.3	77.1	67.7	78.4	79.4
UCF (MO-MP-CNN)	82.5	88.7	93.2	87.4	88.4	77.2	78.4	85.0
CUHK1 (TL+DS+mDSC)	66.54	74.62	86.80	86.16	71.43	71.24	70.19	75.19
GUT (Seg-CNN)	87.7	75.2	77.7	81.1	82.7	76.6	72.0	78.7
ViTSegment (Our Proposal)	**90.27**	**92.66**	**93.26**	**92.91**	**91.95**	**91.45**	**87.97**	**91.50**

Table 5.2: *Dice Similarity Coefficient (DSC) on MRI Dataset from other studies on MICCAI challenge. Participants: University or Research team (proposed method) [40]*

Despite the good performance of the model, this approach has limitations. Firstly, the small dataset used for training and validation, had an significant impact mainly on model's performance on MRI data. Then resource constraints during training, such as reliance on a small GPU, which occur on the use of a minimal batch size of 1, affecting optimization and training efficiency. Additionally, the model is not designed to process both datasets simultaneously due to the different features of them. To address the limitations mentioned above, several future work should be explored. First, training the model with bigger dataset so it can improve generalizability. Then, multi-modal integration could be added to the model so it can train and learn from both CT and MRI datasets. Finally, the current work is a supervised learning model so a future research could focus on exploring the potential of self-supervised or semi-supervised training. The latter could reduce the reliance on manually labeled data, making the model applicable in datasets lacking labeled data.

5.1 Conclusion

In this thesis it was presented an implementation for automated whole-heart segmentation, utilizing a hybrid architecture, called ViTSegment. ViTSegment is a Vision Transformer-based encoder for enhanced feature extraction and a convolutional decoder for accurate boundary detection. The dataset that was used for the training, validation and testing of the model consisted of 20 MRI and 20 CT volumes and their manually segmented label from MICCAI Multi-Modality Whole Heart Segmentation (MM-WHS) challenge. The proposed model effectively outperformed traditional CNN-based models (U-Net, UNETR) in segmentation accuracy, which were trained on the same dataset, as well as the architectures implemented by the participants of the MM-WHS challenge. While the results of ViTSegment are highly promising, challenges still remain, such as precise boundary delineation of smaller structures of the heart, like the Pulmonary Artery, or the weak performance on MRI images. These challenges might have been caused by the limited dataset size that was used for the training. Therefore, access to a larger labeled dataset could potentially limit these issues. Further research could focus maybe on extending the application of ViTSegment to other organ systems and noticing its efficiency for different shaped regions, improving the model's ability to generalize.

Bibliography

[1] Samira Abnar, Mostafa Dehghani, and Willem Zuidema. Transferring inductive biases through knowledge distillation. https://arxiv.org/abs/2006.00555, 2020.

[2] Firas Bachay and Mohammed Abdulameer. Hybrid deep learning model based on autoencoder and cnn for palmprint authentication. *International Journal of Intelligent Engineering and Systems*, 15:2022, 12 2022.

[3] Achraf Ben-Hamadou, Oussama Smaoui, Ahmed Rekik, Sergi Pujades, Edmond Boyer, Hoyeon Lim, Minchang Kim, Minkyung Lee, Minyoung Chung, Yeong-Gil Shin, Mathieu Leclercq, Lucia Cevidanes, Juan Carlos Prieto, Shaojie Zhuang, Guangshun Wei, Zhiming Cui, Yuanfeng Zhou, Tudor Dascalu, Bulat Ibragimov, Tae-Hoon Yong, Hong-Gi Ahn, Wan Kim, Jae-Hwan Han, Byungsun Choi, Niels van Nistelrooij, Steven Kempers, Shankeeth Vinayahalingam, Julien Strippoli, Aurélien Thollot, Hugo Setbon, Cyril Trosset, and Edouard Ladroit. 3dteethseg'22: 3d teeth scan segmentation and labeling challenge. https://arxiv.org/abs/2305.18277, 2023.

[4] Britannica. Blood vessel. https://www.britannica.com/science/human-cardiovascular-system.

[5] M. Jorge Cardoso, Wenqi Li, Richard Brown, Nic Ma, Eric Kerfoot, Yiheng Wang, Benjamin Murrey, Andriy Myronenko, Can Zhao, Dong Yang, Vishwesh Nath, Yufan He, Ziyue Xu, Ali Hatamizadeh, Andriy Myronenko, Wentao Zhu, Yun Liu, Mingxin Zheng, Yucheng Tang, Isaac Yang, Michael Zephyr, Behrooz Hashemian, Sachidanand Alle, Mohammad Zalbagi Darestani, Charlie Budd, Marc Modat, Tom Vercauteren, Guotai Wang, Yiwen Li, Yipeng Hu, Yunguan Fu, Benjamin Gorman, Hans Johnson, Brad Genereaux, Barbaros S. Erdal, Vikash Gupta, Andres Diaz-Pinto, Andre Dourson, Lena Maier-Hein, Paul F. Jaeger, Michael

Baumgartner, Jayashree Kalpathy-Cramer, Mona Flores, Justin Kirby, Lee A. D. Cooper, Holger R. Roth, Daguang Xu, David Bericat, Ralf Floca, S. Kevin Zhou, Haris Shuaib, Keyvan Farahani, Klaus H. Maier-Hein, Stephen Aylward, Prerna Dogra, Sebastien Ourselin, and Andrew Feng. Monai: An open-source framework for deep learning in healthcare. https://arxiv.org/abs/2211.02701, 2022.

[6] Adrian Celaya, Beatrice Riviere, and David Fuentes. A Generalized Surface Loss for Reducing the Hausdorff Distance in Medical Imaging Segmentation. *arXiv e-prints*, page arXiv:2302.03868, February 2023. https://ui.adsabs.harvard.edu/abs/2023arXiv230203868C.

[7] Rehman A Chaudhry R, Miao JH. Physiology, cardiovascular. [updated 2022 oct 16]. in: Statpearls [internet]. treasure island (fl): Statpearls publishing. https://www.ncbi.nlm.nih.gov/books/NBK493197/, 2024.

[8] Chen Chen, Chen Qin, Huaqi Qiu, Giacomo Tarroni, Jinming Duan, Wenjia Bai, and Daniel Rueckert. Deep learning for cardiac image segmentation: A review. *Frontiers in Cardiovascular Medicine*, 7, 2020.

[9] Chen Chen, Chen Qin, Huaqi Qiu, Giacomo Tarroni, Jinming Duan, Wenjia Bai, and Daniel Rueckert. Deep learning for cardiac image segmentation: A review. *Frontiers in Cardiovascular Medicine*, 7, 2020.

[10] Jieneng Chen, Yongyi Lu, Qihang Yu, Xiangde Luo, Ehsan Adeli, Yan Wang, Le Lu, Alan L. Yuille, and Yuyin Zhou. Transunet: Transformers make strong encoders for medical image segmentation. https://arxiv.org/abs/2102.04306, 2021.

[11] Cleveland Clinic. Body Systems and Organs heart. https://my.clevelandclinic.org/health/body/21704-heart. Accessed: 01/26/2024.

[12] Zhiming Cui, Changjian Li, Nenglun Chen, Guodong Wei, Runnan Chen, Yuanfeng Zhou, Dinggang Shen, and Wenping Wang. Tsegnet: An efficient and accurate tooth segmentation network on 3d dental model. *Medical Image Analysis*, 69:101949, 2021.

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg

Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. https://arxiv.org/abs/2010.11929, 2021.

[14] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. https://arxiv.org/abs/1603.07285, 2018.

[15] Stéphane d'Ascoli, Hugo Touvron, Matthew L Leavitt, Ari S Morcos, Giulio Biroli, and Levent Sagun. Convit: improving vision transformers with soft convolutional inductive biases*. *Journal of Statistical Mechanics: Theory and Experiment*, 2022(11):114005, November 2022.

[16] eNCORD. Confusion matrix. https://encord.com/glossary/confusion-matrix/, 2023.

[17] Geeks for Geeks. Artificial neural networks and its applications. https://www.geeksforgeeks.org/artificial-neural-networks-and-its-applications/, 2024.

[18] Hongyang Gao, Hao Yuan, Zhengyang Wang, and Shuiwang Ji. Pixel deconvolutional networks. https://arxiv.org/abs/1705.06820, 2017.

[19] Yunhe Gao, Mu Zhou, Di Liu, Zhennan Yan, Shaoting Zhang, and Dimitris N. Metaxas. A data-scalable transformer for medical image segmentation: Architecture, model efficiency, and benchmark. https://arxiv.org/abs/2203.00131, 2023.

[20] Ali Hatamizadeh, Yucheng Tang, Vishwesh Nath, Dong Yang, Andriy Myronenko, Bennett Landman, Holger Roth, and Daguang Xu. Unetr: Transformers for 3d medical image segmentation. https://arxiv.org/abs/2103.10504, 2021.

[21] Zhong Hong. Attention mechanisms in deep learning: Enhancing model performance. https://medium.com/@zhonghong9998/attention-mechanisms-in-deep-learning-enhancing-model-performance-32a91006092a, 2023.

[22] Zhong Hong. Adaptive learning rate scheduling: Optimizing training in deep networks. https://medium.com/@zhonghong9998/adaptive-learning-rate-scheduling-optimizing-training-in-deep-networks-14d4f95a45d6, 2024.

[23] Nghi Huynh. Understanding evaluation metrics in medical image segmentation. https://medium.com/@nghihuynh_37300/understanding-evaluation-metrics-in-medical-image-segmentation-d289a373a3f, 2023.

[24] Paul A. Iaizzo. *Handbook of Cardiac Anatomy, Physiology, and Devices*. Humana Totowa, NJ, 2005.

[25] IBM. What is a neural network? https://www.ibm.com/topics/neural-networks.

[26] IBM Jacob Murel Ph.D., Eda Kavlakoglu. What is a confusion matrix? https://www.ibm.com/topics/confusion-matrix, 2024.

[27] Abhishek Jain. Semantic vs instance vs panoptic segmentation. https://medium.com/@abhishekjainindore24/semantic-vs-instance-vs-panoptic-segmentation-b1f5023da39f, 2024.

[28] Sejal Jaiswal. Multilayer perceptrons in machine learning: A comprehensive guide. https://www.datacamp.com/tutorial/multilayer-perceptrons-in-machine-learning, 2024.

[29] Mark Scapicchio Jim Holdsworth. What is deep learning? https://www.ibm.com/topics/deep-learning, 2024.

[30] He Kaiming, Zhang Xiangyu, Ren Shaoqing, and Sun Jian. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification kaiming. *Biochemical and Biophysical Research Communications*, 498:254–261, 2018.

[31] Akash Kesrwani. Multi-head self attention: Short understanding. https://medium.com/@akash.kesrwani99/multi-head-self-attention-short-understanding-e90a34866730, 2023.

[32] Medium lathashreeharisha. Dice coefficient! what is it? https://medium.com/@lathashreeh/dice-coefficient-what-is-it-ff090ec97bda, 2023.

[33] Yann LeCun, Y. Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521:436–44, 05 2015.

[34] Katherine (Yi) Li. How to choose a learning rate scheduler for neural networks. https://neptune.ai/blog/how-to-choose-a-learning-rate-scheduler, 2024.

[35] Zhiyuan Li and Sanjeev Arora. An exponential learning rate schedule for deep learning. https://arxiv.org/abs/1910.07454, 2019.

[36] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of Computer Vision Pattern Recognition*, pages 3431–3440, 06 2015.

[37] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. https://arxiv.org/abs/1411.4038, 2015.

[38] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. https://arxiv.org/abs/1711.05101, 2019.

[39] Théo Martin. A (very short) visual introduction to learning rate schedulers (with code). https://medium.com/@theom/a-very-short-visual-introduction-to-learning-rate-schedulers-with-code-189eddffdb00, 2023.

[40] Mihaela Pop, Maxime Sermesant, Pierre-Marc Jodoin, Alain Lalande, Xiahai Zhuang, and Guang Yang. *Statistical Atlases and Computational Models of the Heart. ACDC and MMWHS Challenges*. Springer Cham, 03 2018.

[41] Rehman A. Rehman I. Anatomy, thorax, heart. [updated 2023 aug 28]. in: Statpearls [internet]. treasure island (fl): Statpearls publishing. https://www.ncbi.nlm.nih.gov/books/NBK470256/, 2024.

[42] Deval Shah. Intersection over union (iou): Definition, calculation, code. https://www.v7labs.com/blog/intersection-over-union-guide, 2023.

[43] Dr. Robert Sweetland. Cardiovascular system review key. https://www.homeofbob.com/health/reviews/circulatorySystem.html.

[44] Minh Tran. Understanding u-net. https://towardsdatascience.com/understanding-u-net-61276b10f360, 2022.

[45] Minh Tran. Understanding u-net. https://towardsdatascience.com/understanding-u-net-61276b10f360, 2022.

[46] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. https://arxiv.org/abs/1706.03762, 2023.

[47] Guoping Xu, Xiaxia Wang, Xinglong Wu, Xuesong Leng, and Yongchao Xu. Development of skip connection in deep neural networks for computer vision and medical image analysis: A survey. https://arxiv.org/abs/2405.01725, 2024.

[48] Yufei Xu, Qiming Zhang, Jing Zhang, and Dacheng Tao. Vitae: Vision transformer advanced by exploring intrinsic inductive bias. https://arxiv.org/abs/2106.03348, 2021.

[49] Zhuliang Yao, Yue Cao, Yutong Lin, Ze Liu, Zheng Zhang, and Han Hu. Leveraging batch normalization for vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 413–422, October 2021.

[50] Xiahai Zhuang, Lei Li, Christian Payer, Darko Stern, Martin Urschler, Mattias P. Heinrich, Julien Oster, Chunliang Wang, Orjan Smedby, Cheng Bian, Xin Yang, Pheng-Ann Heng, Aliasghar Mortazi, Ulas Bagci, Guanyu Yang, Chenchen Sun, Gaetan Galisot, Jean-Yves Ramel, Thierry Brouard, Qianqian Tong, Weixin Si, Xiangyun Liao, Guodong Zeng, Zenglin Shi, Guoyan Zheng, Chengjia Wang, Tom MacGillivray, David Newby, Kawal Rhode, Sebastien Ourselin, Raad Mohiaddin, Jennifer Keegan, David Firmin, and Guang Yang. Evaluation of algorithms for multi-modality whole heart segmentation: An open-access grand challenge. https://arxiv.org/abs/1902.07880, 2019.

[51] Xiahai Zhuang, Kawal Rhode, Reza Razavi, David Hawkes, and Sébastien Ourselin. A registration-based propagation framework for automatic whole heart segmentation of cardiac mri. *IEEE Trans. Med. Imaging*, 29:1612–1625, 09 2010.

[52] Xiahai Zhuang and Juan Shen. Multi-scale patch and multi-modality atlases for whole heart segmentation of mri. *Medical Image Analysis*, page 1, 2016. https://www.sciencedirect.com/science/article/pii/S1361841516000219.

[53] International Business Machines Corporation ("IBM"). What is image segmentation? https://www.ibm.com/topics/image-segmentation.