Michael Staff Larsen

Segmentation of Coronary Arteries using Transformers

Master's thesis in Computer Science
Supervisor: Frank Lindseth
Co-supervisor: Gabriel Kiss

June 2023

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

Michael Staff Larsen

Segmentation of Coronary Arteries using Transformers

Master’s thesis in Computer Science
Supervisor: Frank Lindseth
Co-supervisor: Gabriel Kiss
June 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

Abstract

Coronary Artery Disease (CAD) is a significant health issue worldwide. The condition is traditionally diagnosed with invasive, costly methods such as Invasive Coronary Angiography (ICA) and invasive Fractional Flow Reserve (FFR) measurements. These procedures, however, carry associated risks. As a result, there's been a shift towards using the safer, more cost-effective Coronary Computed Tomography Angiography (CCTA), a non-invasive imaging technique. Recent years have seen growing research interest in boosting CCTA's diagnostic potential via automated coronary artery segmentation.

This thesis focused on evaluating the performance of Shifted Window U-Net Transformer (SWIN UNETR), a transformer-based architecture, and contrasting it with current Convolutional Neural Network (CNN)-based methods like no new U-Net (nnU-Net) for coronary artery segmentation. Our experiments revealed that the SWIN UNETR model surpassed previous benchmarks with a Dice Score (DSC) of 0.8614 versus the earlier 0.8296 on the ImageCAS dataset. Moreover, it secured 7th place in the ASOCA Challenge with a competitive DSC of 0.8663. When compared to nnU-Net on the St. Olavs Hospital dataset, SWIN UNETR demonstrated superior performance in terms of DSC and with fewer large artifacts in its predictions.

Furthermore, the integration of automatic coronary artery segmentation with prior FFR estimation work was examined. Although a few areas needed manual corrections, the SWIN UNETR model was successfully used as the input to the FFR estimation method and yielded a strong correlation with physically measured FFR values. Its application in classifying stenosis as functionally significant (FFR < 0.8), demonstrated a promising sensitivity of 85.7% compared to physical measurements. This result exceeded the sensitivity of using clinically segmented arteries as input.

In summary, SWIN UNETR was found to excel at the task of coronary artery segmentation from CCTA images compared to current CNN methods. Additionally, the combination of automatic segmentation and FFR estimation gave promising results when combined with some minor manual corrections. Both the segment-

ation and combination with FFR estimation could therefore be valuable tools for clinical assessment of CAD from CCTA.

As a primer for reading this thesis, it might be beneficial to look at these two demonstration videos: video 1 and video 2. The videos demonstrate the issues with the predicted segmentations made by SWIN UNETR, but they also demonstrate the segmentation task in general.

Sammendrag

Koronar hjertesykdom (CAD) er en betydelig helseutfordring på verdensbasis. Tilstanden diagnostiseres tradisjonelt med invasive, kostbare metoder som Invasive Coronary Angiography (ICA) og invasive Fractional Flow Reserve (FFR)-målinger. Disse prosedyrene medfører imidlertid assosierte risikoer. Som et resultat har det vært en overgang mot å bruke den tryggere, mer kostnadseffektive Coronary Computed Tomography Angiography (CCTA), en ikke-invasiv avbildningsteknikk. De siste årene har det vært økende forskningsinteresse for å øke CCTA's diagnostiske potensial gjennom automatisk segmentering av koronararteriene.

Denne masteroppgaven fokuserte på å evaluere ytelsen til Shifted Window U-Net Transformer (SWIN UNETR), en transformer-basert arkitektur, sammenlignet med nåværende Convolutional Neural Network (CNN)-baserte metoder som no new U-Net (nnU-Net) for segmentering av koronararterier. Våre eksperimenter avdekket at SWIN UNETR-modellen overgikk tidligere resultater med en Dice Score (DSC) på 0.8614 mot 0.8296 på ImageCAS datasettet. Videre sikret SWIN UNETR 7. plass i ASOCA Challenge med en konkurransedyktig DSC på 0.8663. Når den ble sammenlignet med nnU-Net på St. Olavs Hospital datasettet, demonstrerte SWIN UNETR overlegen ytelse i form av DSC og med færre store artefakter i sine prediksjoner.

Videre ble integrasjonen av automatisk segmentering av koronararterier med tidligere FFR estimeringsarbeid undersøkt. Selv om noen områder trengte manuelle korreksjoner, ble SWIN UNETR-modellen vellykket brukt som input til FFR-estimeringsmetoden og ga en sterk korrelasjon med fysisk målte FFR-verdier. Dens anvendelse i klassifisering av stenose som funksjonelt betydelig (FFR < 0.8), viste en lovende følsomhet på 85,7% sammenlignet med fysiske målinger. Dette resultatet overgikk følsomheten ved bruk av klinisk segmenterte arterier som input.

Oppsumert ble det funnet at SWIN UNETR utmerket seg til segmentering av koronararterier fra CCTA-bilder sammenlignet med nåværende CNN-metoder. I tillegg ga kombinasjonen av automatisk segmentering og FFR-estimering lovende resultater når den ble kombinert med noen mindre manuelle korreksjoner. Både segmenteringen og kombinasjonen med FFR-estimering kan derfor være verdifulle verktøy for klinisk vurdering av CAD fra CCTA bilder.

Som en forberedelse til å lese denne masteroppgave, kan det være fordelaktig å se på disse to demonstrasjonsvideoene: video 1 og video 2. Videoene demonstrerer problemene med de predikerte segmenteringene laget av SWIN UNETR, men de demonstrerer også segmenteringsoppgaven generelt.

Preface

Conducted at the Norwegian University of Science and Technology (NTNU) in Trondheim, this Master's thesis is an exploration into the forefront of medical image analysis, particularly focusing on the potential of using transformer models for segmentation of coronary arteries and non-invasive assessment of coronary artery disease.

The thesis evolved within the confines of the NTNU Computer Science department, supported significantly by access to computing hardware essential for the execution of the experiments. An integral part of this work was the Fractional Flow Reserve (FFR) estimation conducted by my co-supervisor Fredrik Eikeland Fossan in the final experiment, a crucial step towards the combination of automatic coronary artery segmentation and FFR estimation for non-invasive assessment of coronary artery disease. Additionally, I would like to thank Fredrik for his guidance in the field of FFR estimation and Coronary Artery Disease (CAD).

Deep gratitude is extended to my supervisor, Frank Lindseth, who not only sparked my interest in medical image analysis but also provided invaluable supervision throughout the execution of experiments and the writing of this thesis. Likewise, the guidance of Gabriel Kiss, my second co-supervisor, has been of significant importance in directing me toward relevant research in this evolving field.

Above all, I wish to thank my wife, family, and friends for their steady support throughout this journey. Lastly, it's my hope that this thesis will make a meaningful contribution to the field of coronary artery disease diagnosis.

Contents

Figures

Tables

Acronyms

AI Artificial Intelligence. xv, 7, 9

ANN Artificial Neural Networks. xv, 2, 7, 8, 28, 29, 42, 43, 60, 61

CAD Coronary Artery Disease. iii, iv, vii, xi, xv, 1–6, 69, 71

CCTA Coronary Computed Tomography Angiography. iii–vi, xv, 1–3, 5, 6, 33, 34, 42, 50, 53, 65, 69, 72

CNN Convolutional Neural Network. iii, v, xv, 2, 3, 5, 9, 18, 19, 41, 65, 71

DICOM Digital Imaging and Communications in Medicine. xv, 33

DL Deep Learning. xv, 2, 5, 7, 8, 14, 24, 64

DSC Dice Score. iii, v, xv, 36, 40, 41, 46–50, 53–56, 63–66, 68

EDA Exploratory Data Analysis. xv

FCNN Fully Connected Neural Network. xv, 8, 9

FFR Fractional Flow Reserve. iii–vii, xi, xv, 1–7, 24, 27–29, 42, 43, 45, 61, 65, 67–69, 71, 72

FPN Feature Pyramid Network. xv, 20

GPU Graphical Processing Unit. xv

ICA Invasive Coronary Angiography. iii, v, xv, 1, 5, 6, 33

IoU Intersect over Union. xv, 9

LMCA Left Main Coronary Artery. xv

NIfTI Neuroimaging Informatics Technology Initiative. xv, 33

NLP Natural Language Processing. xv, 18

nnU-Net no new U-Net. iii, v, x, xiii, xv, 3, 16, 26, 40, 41, 45, 53–56, 66, 68, 71

RCA Right Coronary Artery. xv

RNN Recurrent Neural Networks. xv

ROM Reduced-order model. xv, 2, 29, 43, 60

SOTA State of The Art. xv, 2, 3, 18, 68, 71

SSL Self Supervised Learning. xv, 14, 19, 21, 37, 63, 64

SWIN Shifted Window. xv, 19, 20

SWIN UNETR Shifted Window U-Net Transformer. iii–vi, x, xiii, xv, 20, 21, 36–42, 45–48, 50, 53–56, 59, 61, 63, 64, 66, 68–72

VIT Vision Transformer. xiii, xv, 18–20

Chapter 1

Introduction

1.1 Introduction

Coronary Artery Disease (CAD) is a leading cause of death worldwide, responsible for significant morbidity and mortality [1]. Early detection and accurate assessment of CAD are crucial in guiding therapeutic decisions and improving patient outcomes [2]. CAD is characterized by the degree of flow restriction caused by plaque buildup in the coronary arteries. This plaque buildup is called stenosis when the pressure drop is severe enough. The typical way of diagnosing CAD is with an invasive imaging technique called Invasive Coronary Angiography (ICA). Unless the ICA clearly shows a full, or nearly full blockage, additional pressure measurements are performed and used to calculate the pressure drop across a stenosis. The pressure drop measurement is called Fractional Flow Reserve (FFR) and is calculated using the pressure measured with a sensor at a location before and after a stenosis, with respect to the flow direction. FFR is a valuable metric that can quantify the hemodynamic significance of coronary stenosis, assisting clinicians in deciding whether revascularization is necessary [2].

Although ICA is the gold standard for accurate assessment of CAD, the invasive strategy of ICA is associated with some health risks [3], and is costly to perform [3]. Moreover, approximately 50% of the patients who undergo ICA are found to have non-significant stenosis which does not require intervention. Due to these reasons, non-invasive imaging techniques like Coronary Computed Tomography Angiography (CCTA) are often used as a preliminary visual examination in order to rule out the need for more accurate examination by ICA and FFR measurements [2]. Although CCTA produces lower resolution images, it has the benefit of producing full 3D volumes of the coronary arteries, in contrast to the 2D images produced by ICA. The images from CCTA, are used by radiologists in order to visually assess the extent of potential stenosis in order to guide further action. Often patients can be excluded from needing further action after the CCTA examination, which reduces the number of patients having to undergo invasive examination. However, even with the introduction of CCTA, there are still a lot of unnecessary referrals to ICA. Hence more specific non-invasive diagnostic tools are warranted.

The CCTA images are often examined slice by slice in the 3D domain, which can be a complicated endeavor. A supplementary tool for examining these images is to perform 3D segmentation of the coronary arteries, which makes it easier to inspect. One way of producing this 3D segmentation is to utilize medical 3D image-based software like mimics [4], 3D slicer [5] or ITK Snap [6, 7]. Unfortunately, this is a tedious task and has to be performed by a skilled radiologist. One potential issue with using radiologists for this segmentation task is that humans can make mistakes, which leads to inconsistent segmentation results.

As an alternative to manually segmenting the coronary arteries, Deep Learning (DL) techniques have shown remarkable success in various medical image segmentation tasks, including the segmentation of coronary arteries [8, 9]. State of The Art (SOTA) methods, such as the U-Net and its variants like the 3D U-Net and nnU-Net, have achieved good performance in this domain [10–13]. However, Transformers have recently shown promising results in various computer vision tasks, outperforming traditional Convolutional Neural Network (CNN) based models [14–16]. By leveraging the self-attention mechanism, transformers can capture long-range dependencies and model complex spatial relationships, which can be particularly beneficial for the segmentation of intricate structures like coronary arteries.

Recent work has shown that it is possible to estimate FFR from a 3D segmentation of the coronary arteries using a combination of a physics-based Reduced-order model (ROM) and data-driven Artificial Neural Networks (ANN), with promising results [17]. The authors managed to estimate FFR with a prediction standard deviation error of 0.021 with respect to solving the 3D incompressible Navier–Stokes (iNS) equations. In comparison, the standard deviation of repeated FFR measurements is 0.018.

Combining the use of DL based segmentation and FFR estimation could potentially be used to create a fully automated pipeline for estimating FFR directly from CCTA images. If this technique is viable, it has the potential to be a faster, cheaper and safer alternative to the traditional invasive strategy of diagnosing CAD. In order for this pipeline to work robustly, the quality of the segmentation is paramount.

1.2 Goal and Research Questions

The primary aim of this master's thesis is to train and assess a SOTA transformer-based model for segmentation of coronary arteries. To thoroughly evaluate the efficacy of the transformer-based model, its results will be compared to prior work on two publicly available datasets [8, 9] and a private dataset from St. Olavs Hospital, located in Trondheim, Norway. The three datasets are consisting of CCTA volumes and quality assured segmentations of coronary arteries. As there are no public benchmark on the St. Olavs Hospital dataset, the transformer model is compared to a no new U-Net (nnU-Net) model trained on the same dataset. Beyond segmentation, this thesis aims to explore the potential integration of the segmentation model with existing work on the estimation of FFR.

Research question 1: How do recent transformer-based architectures compare with current CNN-based methods for segmentation of coronary arteries?

Research question 2: Is it possible to combine automatic coronary artery segmentation with previous work on FFR estimation for clinical assessment of CAD?

1.3 Research Method

This thesis employs an experimental research strategy complemented by a literature review. To address the first research question, experiments are conducted comparing a SOTA transformer-based architecture with CNN-based methods for coronary artery segmentation. This involves hyper-parameter adjustments for the transformer model to optimize its accuracy.

The second question is tackled by experimentally combining automatic coronary artery segmentation with prior work on FFR estimation to improve non-invasive CAD assessments and reduce user dependence. The integration's effectiveness is assessed through further trials, supported by insights from the literature review on FFR estimation and automatic segmentation.

1.4 Contributions

Transformer-based networks have shown powerful abilities in various segmentation tasks, but to the best of our knowledge have not been tested on segmentation of coronary arteries. Hence, the main contribution of this thesis is therefore a thorough exploration and comparison of a transformer-based alternative to CNN based SOTA methods.

The availability of a recently published dataset called ImageCAS, which comprises approximately 1000 samples [9] of coronary artery segmentations, presented an opportunity to evaluate the performance of the architecture when trained on a significantly larger dataset compared to previously available public datasets [8].

Another contribution is the research done in exploring the possibility of combining previous work on estimation of FFR and automatic segmentation, in order to make

FFR estimation for clinical assessment of CAD less user-dependent.

1.5 Thesis Outline

This thesis is structured into the following chapters:

Chapter 1 - Introduction: Presents the motivation for the study, the goal and research questions as well as the contributions of this thesis.

Chapter 2 - Background and Related Work: Introduces the necessary theory about CAD, deep learning, computer vision and 3D segmentation of coronary arteries. Finally, a selection of related work of importance to segmentation of coronary arteries and FFR estimation will be presented.

Chapter 3 - Methodology: Presents the tools, datasets and methodology for the experiments for coronary artery segmentation and FFR estimation.

Chapter 4 - Results: Presents the quantitative and qualitative results from the experiments performed.

Chapter 5 - Discussion: Discusses the implications of the results and relates the findings to the research questions.

Chapter 6 - Conclusion and Future Work: Summary of the key findings, conclusion and suggestions for further work.

Chapter 2

Background and Related Work

This chapter presents the background material and related work of this thesis. First, relevant aspects of CAD, the imaging techniques, and FFR are explained. Followed by an introduction to DL and computer vision for medical image segmentation. In addition, a selection of CNNs and transformer models are explained. Finally a selection of related work of special interest to the subject of performing semantic segmentation of coronary arteries, as well as work on estimating FFR is presented.

2.1 Coronary Artery Disease

CAD is a condition characterized by the narrowing or blockage of the coronary arteries, which supply blood to the heart muscle. This narrowing is due to the buildup of plaque, a combination of fat, cholesterol, calcium, and other substances found in the blood, see Figure 2.1. When the coronary arteries become partially or completely blocked, blood flow to the heart muscle is reduced, potentially leading to angina (chest pain) or even a heart attack. CAD is a leading cause of death worldwide and accounts for a significant number of hospitalizations and healthcare costs [1].

2.1.1 Diagnosis

The diagnosis of CAD involves a variety of imaging techniques and functional tests, aimed at assessing the degree of stenosis (narrowing) in the coronary arteries, as well as the hemodynamic impact of this narrowing on blood flow to the heart muscle [2]. Here we describe three commonly used diagnostic tools: ICA, CCTA, and FFR).

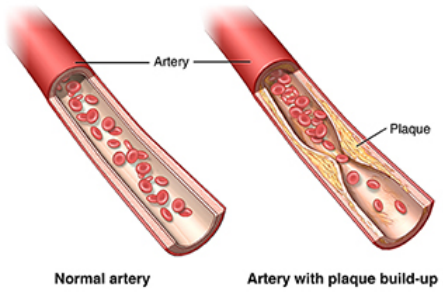


Figure 2.1: Plaque buildup illustration

Source: [18]

Invasive coronary angiography

ICA involves the insertion of a catheter through an artery, usually in the groin or wrist, and advancing it to the coronary arteries under X-ray guidance [19]. Once the catheter is in place, a contrast agent is injected into the coronary arteries, and X-ray images are taken during the passage of the contrast agent. These X-ray images, called angiograms, provide a two-dimensional visualization of the coronary arteries, enabling the identification of stenosis or other abnormalities.

Despite its advantages in providing accurate and detailed information about the coronary arteries, ICA is an invasive procedure and carries certain risks, such as bleeding, infection, allergic reactions to the contrast agent, and radiation exposure [3]. Additionally, ICA is expensive and time-consuming, which makes it less suitable for widespread screening of CAD. As a result, noninvasive imaging techniques, such as CCTA, have been developed and are increasingly used to complement or replace ICA in certain clinical scenarios [2].

Computed tomography coronary angiography

CCTA is a non-invasive imaging technique that utilizes a CT scan in conjunction with a contrasting fluid [19]. The contrasting fluid is injected intravenously and allows the inner part of the coronary arteries (lumen) to be contrasted. The resulting image from this technique has three dimensions and can be used to visually conclude if the patient has a severe stenosis.

Fractional flow reserve

FFR is a measure of the pressure difference across a coronary artery stenosis, used to determine the functional significance of the narrowing [20]. FFR is typically obtained during an ICA procedure by inserting a pressure-sensitive guidewire across the stenosis. The pressure measurements are taken before (proximal) and after (distal) the stenosis and are used to calculate the FFR value, see Equation (2.1).

An FFR value of less than 0.80 indicates a functionally significant stenosis, which may require the insertion of a tube (stent) that expands the arteries in order to regain the lost flow, also known as revascularization [2].

$$FFR = \frac{\bar{P}_{distal}}{\bar{P}_{proximal}} \tag{2.1}$$

2.1.2 Data representation and Hounsfield Units

CCTA images are represented in a three-dimensional grid of voxels (pixels in 3D space), where each voxel contains a numerical value representing the radiodensity of that specific region of the scanned anatomy. These voxel values are stored and displayed in Hounsfield Units, a quantitative scale for describing radiodensity, widely used in computed tomography. The HU scale was introduced by Godfrey Hounsfield, the inventor of the CT scanner, and it has become the standard in radiological practice [21].

In the Hounsfield scale, the radiodensity of distilled water at standard temperature and pressure is defined as zero HU, while the radiodensity of air is defined as negative 1000 HU. Radiodensities greater than water are assigned positive HU values, and those less than water are assigned negative values. For instance, the typical HU values for various tissues are approximately +45 HU for blood and +1000 HU for bone [21].

In the context of CCTA images, the coronary artery structure can be identified by the radiodensity of the contrastive fluid, in contrast to the surrounding anatomy. The radiodensity can also be used to identify plaque built up inside the arteries. Unfortunately, typical radiodensities of plaque and lumen (contras fluid) have some overlap. Plaque radiodensity can be divided into several components depending on the content: soft plaque (-100-29HU), fibrous plaque (30-149HU) and calcified plaque (150-1300HU) [22]. In comparison, the radiodensity of coronary arteries ranges from 200-500HU [23]. Note that the HU values may vary depending on the specific CT scanner and imaging protocol used.

2.2 Deep Learning

DL is a subcategory of Artificial Intelligence (AI), and DL are built on top of the idea of artificial neurons and ANN. An artificial neuron is based on a biological neuron and can be defined as in Equation (2.2). Here x_i is the input, w_i is the weights and is then added with a bias after the summation. This is usually passed into an activation function like sigmoid, in order to introduce non-linearity [24].

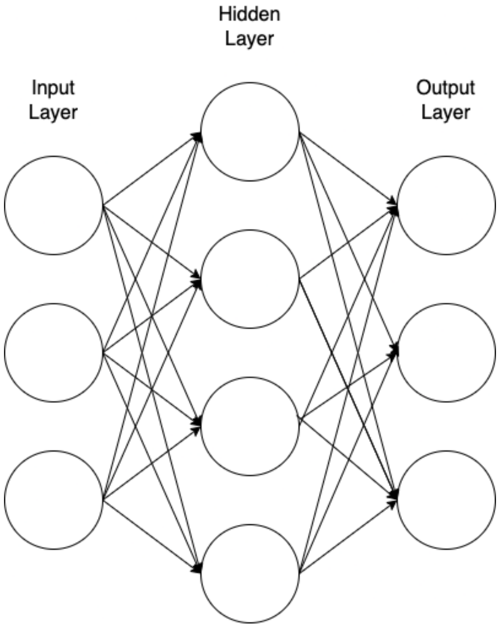


Figure 2.2: Simple artificial neural network architecture
This network has 3 input nodes in the input layer, 4 nodes in the hidden layer and 3 output nodes in the output layer. Notice that every node from each layer is connected to every node in the next layer, making this network a fully connected neural network (FCNN).

$$\sum_{i=1}^{m} w_i x_i + bias \tag{2.2}$$

A common form of ANN is Fully Connected Neural Network (FCNN). In a FCNN all the outputs of one layer is connected to every neuron in the next layer. DL is defined as an ANN that has 3 or more layers including the input and output layer. See Figure 2.2 for an illustration of a simple ANN that would also classify as DL and FCNN.

The training regimen for an ANN commences with a forward pass, where the network's outputs are computed based on the given inputs. Subsequently, these outputs are compared with the corresponding target values, and a loss is determined based on the divergence between the predicted and actual values. The gradient of this loss function, with respect to the weights in each layer, is calculated using partial differentiation, a process that provides insight into how small changes in the weights might influence the loss. This gradient information is then utilized to iteratively adjust the weights and biases in the network in a direction that minimizes the output of the loss function. This iterative optimization process is known as gradient descent, and it serves to incrementally improve the predictive performance of the ANN [24].

2.3 Computer Vision

Computer vision is a popular usage of deep learning, and has mostly been dominated by CNN. CNN uses convolutional filters with trainable parameters in order to extract features from an image. The filters can also downsample or upsample the image at different layers in order to extract information of various sizes. This allows for a lower computational cost than using FCNN. Features in a CNN are learned, and range from simple features in the lower layers to more complex features in the deeper layers.

2.3.1 Classification

One of the typical tasks in AI is classification. The task consists of assigning the input data to a given class of the output domain. A classic example from computer vision is, given a set of images of either a dog or a cat, assign the image to the correct class.

2.3.2 Object Detection

Object detection localizes an object within an image. The localization is done by predicting coordinates for a bounding box around the object.

A common metric for validating object detection is the Intersect over Union (IoU) [25]. IoU works by calculating the area where the prediction overlaps the ground truth label and then dividing by the union area of both (Figure 2.3). If the prediction is perfect, the IoU would be 1.

2.3.3 Semantic Segmentation

Semantic segmentation is an extension of classification, but instead of classifying the whole image as a class, a classification is performed for each pixel [25]. This task falls under the category of dense prediction.

Semantic segmentation can also be used for 3D volumes, like segmentation of organs. In 3D volumes, each voxel (pixel in 3D space) is assigned to a class, see Figure 2.4.

2.3.4 Validation Metrics

Proper evaluation of the performance of semantic segmentation models is crucial for comparing different approaches and determining their suitability for specific tasks. Several validation metrics have been proposed to assess the quality

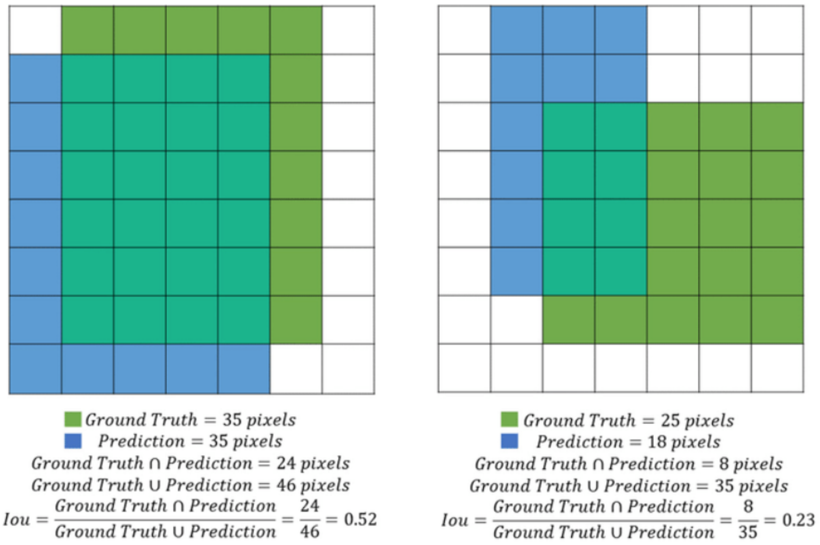

🟩 Ground Truth = 35 pixels
🟦 Prediction = 35 pixels
Ground Truth ∩ Prediction = 24 pixels
Ground Truth ∪ Prediction = 46 pixels
$Iou = \frac{Ground\ Truth \cap Prediction}{Ground\ Truth \cup Prediction} = \frac{24}{46} = 0.52$
🟩 Ground Truth = 25 pixels
🟦 Prediction = 18 pixels
Ground Truth ∩ Prediction = 8 pixels
Ground Truth ∪ Prediction = 35 pixels
$Iou = \frac{Ground\ Truth \cap Prediction}{Ground\ Truth \cup Prediction} = \frac{8}{35} = 0.23$

Figure 2.3: This figure illustrates the Intersect over union calculation on the pixel level, with two examples.

Source: [26]

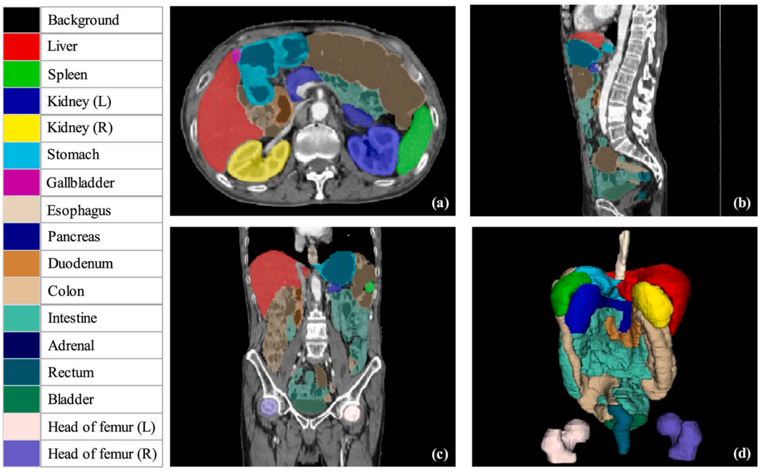


Figure 2.4: Semantic segmentation in 3D

Source: [27]

of segmentation results, taking into account various aspects such as region-based similarity, boundary-based similarity, and spatial distance between segmented regions. In this section, we discuss two widely used validation metrics in semantic segmentation: the Dice coefficient and the Hausdorff distance.

Dice Coefficient

The Dice coefficient, also known as the Sørensen-Dice coefficient or the F1 score, is a region-based similarity metric that measures the overlap between two binary segmentation masks. It is defined as the ratio of twice the intersection of the predicted segmentation mask and the ground truth mask to the sum of the number of pixels in both masks:

$$\text{Dice}(\mathbf{P}, \mathbf{G}) = \frac{2 \cdot |\mathbf{P} \cap \mathbf{G}|}{|\mathbf{P}| + |\mathbf{G}|}, \tag{2.3}$$

where **P** is the predicted segmentation mask, **G** is the ground truth mask. The Dice coefficient ranges from 0 to 1, with 1 indicating a perfect overlap and 0 indicating no overlap [28].

The Dice coefficient is widely used in medical image segmentation due to its robustness to class imbalance and its ability to assess the agreement between the predicted and ground truth masks. However, it does not account for the spatial distance between segmented regions, which may be important in some applications [29].

Hausdorff Distance

The Hausdorff distance is a boundary-based metric that measures the spatial distance between the boundaries of the predicted segmentation mask and the ground truth mask. It is defined as the maximum of the minimum distances between points in the two boundaries:

$$d_H(A, B) = \max\left\{\max_{a \in A} \min_{b \in B} d(a, b), \max_{b \in B} \min_{a \in A} d(a, b)\right\} \tag{2.4}$$

In this equation, $d_H(A, B)$ is the Hausdorff distance between two point sets A and B, $d(a, b)$ is the Euclidean distance between points a and b. The Hausdorff distance is sensitive to the spatial arrangement of the segmented regions and can capture local discrepancies between the predicted and ground truth boundaries [30].

While the Hausdorff distance provides complementary information to the Dice coefficient, it is sensitive to outliers and can be influenced by a few large distances. To mitigate this issue, the average or percentile Hausdorff distances can be used as alternative metrics [30]. The Dice coefficient and Hausdorff distance are two widely used validation metrics in semantic segmentation, each providing unique insights into the performance of the segmentation models. While the Dice coefficient focuses on region-based similarity, the Hausdorff distance captures boundary-based discrepancies. Combining these metrics can offer a more comprehensive evaluation of segmentation models, particularly in medical imaging applications.

2.3.5 Loss Functions

Semantic segmentation requires the careful selection of loss functions to guide the optimization of deep learning models. This section will provide an overview of several popular loss functions used in semantic segmentation, including Dice Loss, Soft Dice Loss, Cross-Entropy Loss, Focal Loss, and common combinations of these functions.

Dice Loss

The Dice loss is computed as the complement of the Dice coefficient, see Equation (2.3). Although this loss function is a common choice for 3D segmentation, it is not fit for backpropagation without modifications. The issue occurs due to the inputs being represented discretely as binary masks and are thus not differentiable.

$$L_{Dice} = 1 - DSC(P, G) \tag{2.5}$$

Soft Dice Loss

Soft Dice Loss is a smooth and differentiable variant of the Dice Loss, suitable for backpropagation, as the inputs are the probabilities for each predicted pixel or voxel. [31]. It is computed as follows:

$$L_{SoftDice} = 1 - \frac{2\sum_{i=1}^{N} p_i g_i + \epsilon}{\sum_{i=1}^{N} p_i + \sum_{i=1}^{N} g_i + \epsilon} \tag{2.6}$$

where p_i and g_i denote the predicted probabilities and the ground truth class for each voxel i, N is the number of voxels, and ϵ is a small constant to avoid division by zero. Note that Soft Dice loss is often just called Dice loss.

Cross-Entropy Loss

Cross-Entropy loss is often used in classification tasks and it measures the negative log likelihood of predicting correct labels [24]. The Cross-Entropy loss is defined as:

$$L_{CE} = -\sum_{i=1}^{N} [g_i \log p_i + (1 - g_i) \log(1 - p_i)] \tag{2.7}$$

where p_i and g_i denote the predicted and ground truth probabilities for each voxel i, respectively, and N is the number of voxels.

Focal Loss

Focal loss, introduced by Lin et al. [32], addresses the issue of class imbalance in semantic dense prediction tasks. Focal loss modulates the Cross-Entropy loss with a scaling factor that down-weights the contribution of predictions with high certainty, which allows the loss function to focus on the predicted area with more uncertainty. The focal loss is defined as:

$$L_{Focal} = -\sum_{i=1}^{N} \alpha_i \left[g_i (1 - p_i)^\gamma \log p_i + (1 - g_i)(p_i)^\gamma \log (1 - p_i) \right] \tag{2.8}$$

where α_i is a weighting factor, γ is the focusing parameter, and p_i and g_i represent the predicted and ground truth probabilities for each voxel i.

Soft Dice Loss in combination with other loss functions

The benefit of combining Soft Dice Loss with other loss functions in semantic segmentation tasks comes from their complementary strengths. Soft Dice Loss, which effectively handles class imbalance and provides a broad perspective by considering both the correctly and incorrectly predicted pixels, works well in tandem with other loss functions. Each loss function has unique benefits and handles certain aspects of prediction errors. By integrating Soft Dice Loss with other functions, a more comprehensive loss function is created, better tailored to address the challenges specific to each segmentation task. This comprehensive approach enhances the model's robustness and accuracy, leading to improved performance in semantic segmentation tasks.

Soft Dice Loss + Cross-Entropy Loss: Combining Soft Dice Loss with Cross-Entropy Loss allows the model to leverage the benefits of both global (Soft Dice Loss) and local (Cross-Entropy Loss) information. Soft Dice Loss is less sensitive to false positives and false negatives, whereas Cross-Entropy Loss penalizes misclassifications more heavily. This combination can improve the model's ability to handle class imbalance and small object detection, as well as produce more accurate segmentation results. The combined loss function is defined as:

$$L_{SoftDice+CE} = \alpha L_{SoftDice} + \beta L_{CE} \tag{2.9}$$

where α and β are the weighting factors for Soft Dice Loss and Cross-Entropy Loss, respectively.

Soft Dice Loss + Focal Loss: Combining Soft Dice Loss with Focal Loss allows the model to leverage the global information from Soft Dice Loss while addressing class imbalance issues with the Focal Loss. Focal Loss modulates the Cross-Entropy Loss with a scaling factor that down-weights easy examples and focuses on harder examples, making it more robust against class imbalance [32]. The combination can lead to improved segmentation performance, especially for cases with significant class imbalance. The combined loss function is defined as:

$$L_{SoftDice+Focal} = \alpha L_{SoftDice} + \beta L_{Focal} \tag{2.10}$$

where α and β are the weighting factors for Soft Dice Loss and Focal Loss, respectively.

Selecting an appropriate combination of loss functions can greatly impact the performance of the model, helping to address challenges such as class imbalance and small object detection.

Self Supervised Learning

In many domains there is a lack of data available in order to properly train DL models. In the medical field, labeled data is scarce due to the time-consuming endeavor of labeling this data. Another factor limiting the supply of labeled medical images is that there are heavy regulations on what is allowed to be distributed, due to privacy laws. One clever way to tackle this problem is to perform Self Supervised Learning (SSL) as a preliminary pre-training.

In contrast to supervised learning, where the training data contains labeled targets. SSL can be trained to learn a representation of the data without providing labeled targets. The learning comes from looking at the data and posing it as if it was a supervised learning problem.

Given a dataset $[\hat{x}_i, x_i]_{i=1}^{N}$ learn a function f that maps $\hat{x}_i$ to x_i.

An example from computer vision (CV) is to mask out patches of an image, then try to reconstruct the original image. This is the case in Masked Autoencoders, where the encoder learns a representation and the decoder predicts the missing patches of the image [33].

In this example, the goal is to have an encoder that has learned a representation of the data. This encoder can then be used as a pre-trained encoder in addition to a decoder for downstream tasks like semantic segmentation. This new network is then trained through supervised learning which is called fine-tuning. This method has been shown to increase the model accuracy without adding additional training data [33].

Transfer Learning

Transfer learning is a technique that leverages the knowledge learned from one task or domain and applies it to a different but related task or domain. In computer

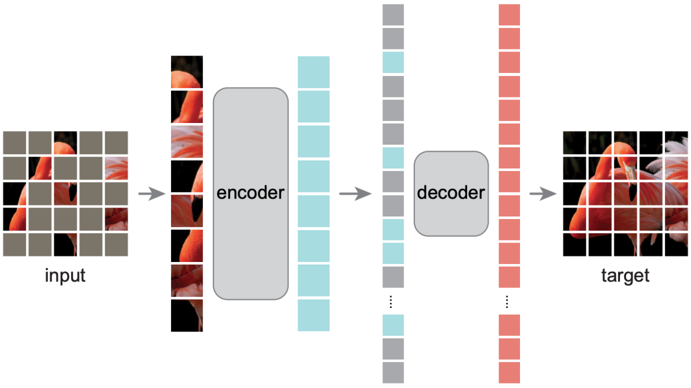


Figure 2.5: How MAE reconstructs a masked image

Source: [33]

vision, transfer learning is often used to address the issue of limited labeled data, especially in medical imaging, where obtaining annotations can be expensive and time-consuming [34].

The most common approach in transfer learning is to pre-train a model on a large labeled dataset, such as ImageNet [35], and then fine-tune the model on the target task using a smaller labeled dataset. This process assumes that the learned features from the source dataset are general enough to be useful for the target task. Fine-tuning can involve updating the entire model or only specific layers, such as the layers in the classifier head. This strategy has been shown to improve performance and convergence speed in various tasks.

2.4 CNN Architectures

2.4.1 U-Net

U-Net is a popular convolutional neural network (CNN) architecture specifically designed for biomedical image segmentation [36]. The architecture consists of an encoding path that captures context and a decoding path that enables precise localization, see Figure 2.6. U-Net has demonstrated exceptional performance in various medical image segmentation tasks, making it a popular choice in the field.

2.4.2 3D U-Net

To address the challenges of volumetric data, such as 3D medical images, a 3D variant of the U-Net architecture has been proposed [13]. The 3D U-Net extends the original 2D U-Net by using 3D convolutions instead of 2D convolutions, allowing it to process volumetric data directly. This adaptation enables the 3D U-Net to

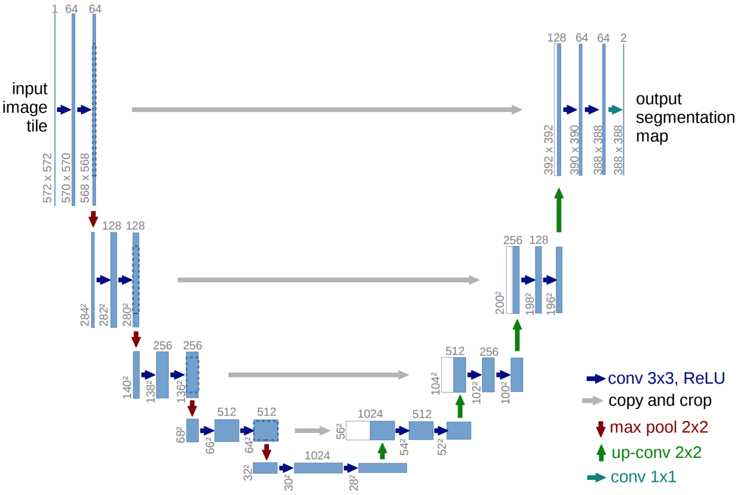


Figure 2.6: U-Net architecture

Source: [10]

better exploit the spatial information in all three dimensions, leading to improved segmentation results for volumetric data.

2.4.3 nnU-Net

nnU-Net is a recent framework built upon the U-Net architecture, which automates the process of designing and configuring CNNs for medical image segmentation [11]. The framework introduces several improvements to the original U-Net, such as automatic network architecture selection and hyper-parameter optimization. These improvements has enabled nnU-Net to achieve state-of-the-art performance in various medical image segmentation tasks without the need for manual architecture design or hyper-parameter tuning.

In conclusion, U-Net and its variants, such as 3D U-Net and nnU-Net, have demonstrated remarkable success in medical image segmentation tasks. Their ability to capture context and localize precisely, along with their adaptability to handle different types of data, make them essential tools in the field of medical image analysis.

2.5 Transformer Architectures

The transformer architecture finds its origins in Recurrent Neural Networks (RNNs). Although it lacks the recurrent nature of RNNs, it shares the attention mechanism. This mechanism was first introduced in the seminal paper "Attention is All You Need" as an alternative to RNNs for natural language processing (NLP) tasks [37]. The new approach enables parallel processing, which significantly improves GPU utilization and allows for faster training, even with a larger number of parameters. As a result, the network can be much larger than previous methods, which in turn enhances the model's capacity. Transformers are better equipped to capture long-range dependencies compared to RNNs, as their self-attention mechanism can directly model relationships between distant elements in the input sequence [37]. Since the release of the "Attention is All You Need" paper, transformer-based architectures have been state-of-the-art in various NLP tasks [38].

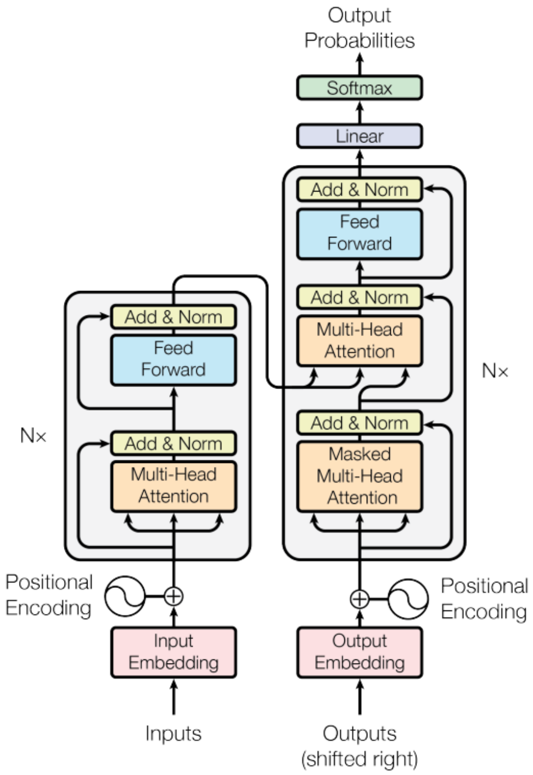


Figure 2.7: Transformer architecture

Source: [37]

The core component of transformer networks is the self-attention mechanism. Given an input sequence, the self-attention mechanism calculates the relationships between different elements of the sequence. The mechanism uses queries (Q), keys (K), and values (V), which are derived from the input x and multiplied by the corresponding trainable weights W^q, W^k and W^v.

The Scaled Dot-Product Self-Attention mechanism, as utilized most transformer networks, can be expressed as the following equation:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2.11}$$

QK^T calculates the dot product between the query and the key, effectively measuring the compatibility between every pair of query and key. The dot product is divided by $\sqrt{d_k}$, where d_k is the dimensionality of the key vectors. This 'scaling' step helps in stabilizing the range of the dot products, especially when d_k is large. The softmax function is applied to ensure the output values are in the range of 0 to 1, and they sum to 1. This step transforms the compatibility scores into 'attention scores' that can be interpreted as probabilities. Finally, these attention scores are used to take a weighted sum of the value vectors, V. This means that values associated with higher scoring keys will contribute more to the final output, hence the name 'attention'.

This mechanism allows the model to attend to different parts of the input sequence to different degrees, thus focusing on the most relevant parts for a given task.

The transformer architecture also utilizes multi-headed self-attention, which consists of multiple self-attention blocks concatenated at the output. This allows the different self-attention blocks to attend to various types of relationships in the input, improving the model's ability to make accurate predictions. Additionally, there is a positional encoding added to the input in order to keep track of the spatial relationship between the different parts of the input. The positional encoding is important, as the input is processed in parallel, see Figure 2.7 for an illustration of the building blocks of the full network architecture.

2.5.1 Vision Transformer

In 2020, researchers at Google realized that the transformer architecture could also be used on images. Their success led to their paper called "A Picture is Worth 16x16 Words" [14]. As the name implies, an image is split up into patches of 16 by 16 pixels, and then flattened to become similar to the word embeddings used for Natural Language Processing (NLP), see Figure 2.8. In this paper, they managed to get slightly better results than the previous SOTA methods for image classification, when pre-trained on a massive dataset (JFT-300M). This model did however not perform better when pre-trained on smaller datasets. One of the improvements that the Vision Transformer (VIT) has, is the size of the attended area in the heads of the network. In the paper, they showed that even in the lower heads, the network attends to a large portion of the image. This is different from CNNs where the receptive field is limited to smaller patches in the lower layers [39].

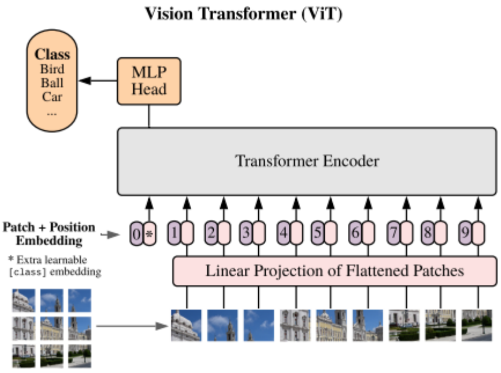

Vision Transformer (ViT)
Class
Bird
Ball
Car
...
MLP Head
Transformer Encoder
Patch + Position Embedding
* Extra learnable [class] embedding
0 * 1 2 3 4 5 6 7 8 9
Linear Projection of Flattened Patches

Figure 2.8: Vision Transformer Architecture

Source: [14]

2.5.2 Improvements to Vision Transformer

Although the VIT paper posed as an interesting alternative to CNNs, the need for a huge dataset did not make it viable for most applications. Especially because the JFT-300M is not publicly available. One way of reducing the need for a big dataset is the usage of SSL, which was successfully implemented with good results in various papers [33, 40, 41].

The VIT paper uses a fairly large patch size which makes it too coarse for dense predictions. The reason for the patch size is the quadratic time complexity ($O(n^2)$). The Shifted Window (SWIN) transformer on the other hand utilizes hierarchical patch sizes and shifting windows [16]. The lower layers of the network start with a much smaller patch size, which allows the network to represent more fine-grained information. This makes the architecture fit for dense prediction tasks like semantic segmentation.

In order to keep the computational cost down, smaller windows of attention are used. This is in contrast to the global attention used in VIT (fig 2.9). The number of patches inside a window is constant, which reduces the time complexity to $O(n)$ with respect to the number of patches.

In addition to the windowed attention block, a shifted windowed attention is used to capture relations between the neighboring windows.

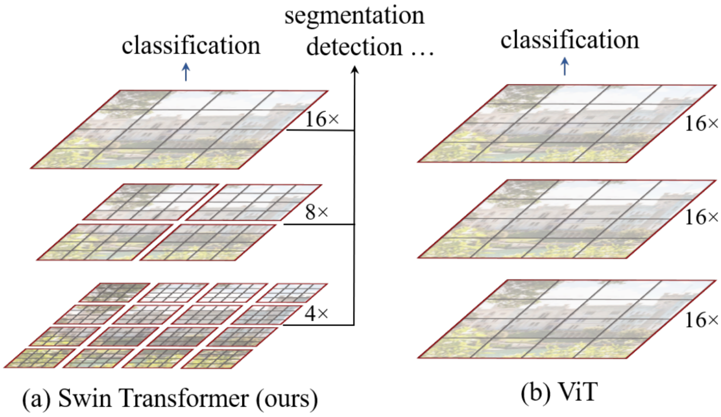


Figure 2.9: Patch and window size in SWIN transformer vs VIT

Source: [16]

2.5.3 Shifted Window U-Net Transformer

The hierarchical feature maps of the SWIN transformer lends itself to be combined with techniques for dense prediction such as Feature Pyramid Network (FPN) [42] and U-Net [10, 16]. This is exactly what the Nvidia research team leveraged by implementing a U-net style decoder in their 2022 paper where they presented SWIN [15]. The decoder comprises of convolutional layers and is connected to each resolution output by skip connections, see Figure 2.10.

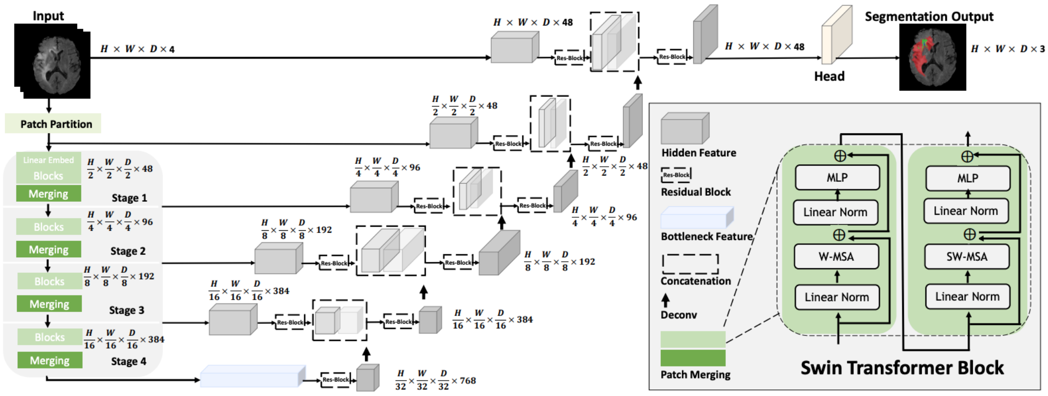

Input
$H \times W \times D \times 4$
Patch Partition
Linear Embed
Blocks
Merging
Stage 1
$\frac{H}{2} \times \frac{W}{2} \times \frac{D}{2} \times 48$
Blocks
Merging
Stage 2
$\frac{H}{4} \times \frac{W}{4} \times \frac{D}{4} \times 96$
Blocks
Merging
Stage 3
$\frac{H}{8} \times \frac{W}{8} \times \frac{D}{8} \times 192$
Blocks
Merging
Stage 4
$\frac{H}{16} \times \frac{W}{16} \times \frac{D}{16} \times 384$
$\frac{H}{32} \times \frac{W}{32} \times \frac{D}{32} \times 768$
Res-Block
$H \times W \times D \times 48$
Head
Segmentation Output
$H \times W \times D \times 3$
Hidden Feature
Residual Block
Bottleneck Feature
Concatenation
Deconv
Patch Merging
MLP
Linear Norm
W-MSA
SW-MSA
Swin Transformer Block

Figure 2.10: Architecture of Shifted Window U-Net Transformer (SWIN UNETR)

Source: [15]

In addition to the U-net decoder, this architecture was also designed for 3D volumes.The patches that are extracted are also in 3D as seen in Figure 2.11

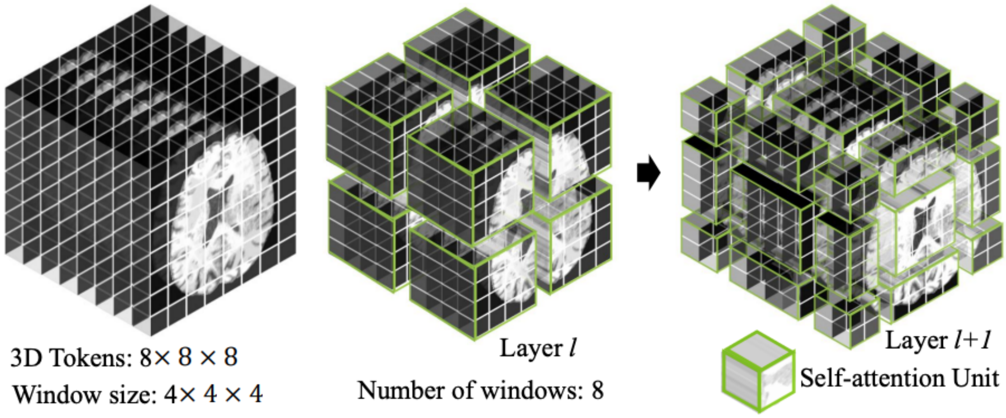


Figure 2.11: SWIN UNETR patches and attention windows

Source: [15]

Self Supervised Learning

Some of the authors of the SWIN UNETR model also released a paper outlining a novel SSL strategy in order to train the SWIN UNETR model [43]. The method samples 3D patches from the source volume and performs two different transformations on the two respective copies.

The first transformation is a random rotation along the z-axis. A multi-layer perceptron (MLP) classification head is then used to predict the correct rotation. The loss function used for this task is a cross entropy loss. The second transformation is a random masking of sections in the sub-volume. The model is then tasked to reconstruct the masked part and is compared with the original sub-volume with a L1 loss function. This technique is inspired by previous work like [33] but modified in order to work with volumetric data. The final part of the method is to use contrastive learning in order to identify the matching augmented sub-volumes in contrast to augmented sub-volumes from other other areas. The contrastive loss is measured using a cosine-similarity. According to the authors, this technique should strengthen the intra-class compactness as well as the inter-class separability. The total loss is calculated as a weighted sum of these three loss functions, see Equation (2.12).

$$L_{tot} = \gamma_1 L_{rotation} + \gamma_2 L_{contrastive} + \gamma_3 L_{reconstruction} \tag{2.12}$$

2.6 Class Imbalance in Medical Imaging

Class imbalance is a common challenge in the field of medical imaging, particularly in tasks such as segmentation and classification. The problem occurs when the distribution of classes in the training dataset is not uniform, with some classes having significantly more examples than others. In medical imaging, the class imbalance can be a result of the rarity of certain diseases or the uneven distribution of anatomical structures within images. This disproportion can lead to biases in the trained model, which may favor the majority class and undermine the performance of the model on the minority class.

2.6.1 Causes of Class Imbalance in Medical Imaging

There are several reasons why class imbalance can occur in medical imaging:

Rare diseases or conditions: Certain diseases or conditions may be relatively rare in the general population, and therefore, fewer examples of these conditions are available in the training dataset. As a result, the model may struggle to learn the patterns associated with these rare conditions, leading to poor performance in detecting or segmenting them.

Uneven distribution of anatomical structures: In medical images, some anatomical structures or regions may be relatively small compared to the overall image size. For example, in the task of segmenting brain tumors, the tumor region may occupy a small portion of the image, while healthy brain tissue occupies the majority. This can cause the model to focus on learning the features of the healthy tissue rather than the tumor, resulting in low sensitivity to the minority class.

Data collection bias: The process of collecting and curating medical imaging datasets can introduce biases that contribute to class imbalance. For instance, some medical centers may have access to a larger number of patients with a specific condition, while others may not. Additionally, the expertise of the radiologists or clinicians involved in data annotation can also impact the distribution of classes in the dataset.

2.6.2 Addressing Class Imbalance in Medical Imaging

Various techniques have been proposed to mitigate the effects of class imbalance in medical imaging:

Data augmentation: Generating additional examples for the minority class through techniques such as rotation, scaling, and flipping can help to balance the class distribution in the training dataset. This approach can improve the model's performance by providing more diverse examples of the minority class.

Resampling: Oversampling the minority class or undersampling the majority class can be used to create a balanced dataset. However, oversampling may lead to overfitting, while undersampling can result in the loss of valuable information from the majority class.

Loss functions: Using specialized loss functions that account for class imbalance, such as Dice loss and Focal loss, and their combinations, can help to optimize the model's performance on imbalanced datasets. These loss functions aim to reduce the influence of the majority class while enhancing the importance of the minority class during training.

In summary, class imbalance is a significant challenge in medical imaging, and various strategies have been proposed to address this issue. By combining these techniques, it is possible to improve the performance of deep learning models on imbalanced medical imaging datasets, leading to more accurate and robust results in clinical applications.

2.7 Morphological Operations

Morphological operations are derived from mathematical morphology. They are particularly useful in image segmentation tasks, which involve partitioning an image into distinct regions corresponding to different anatomical structures or regions of interest. Morphological operations can help refine segmentation results, remove noise, and enhance relevant structures in the images. The fundamental morphological operations are erosion, dilation, opening, and closing, which are typically applied to binary images but can also be extended to grayscale images [44].

2.7.1 Basic Morphological Operations

Erosion: Erosion is an operation that shrinks the boundaries of the foreground regions in a binary image. It is performed using a structuring element, which is a smaller binary matrix that slides over the image. A pixel is set to 1 if all structuring elements overlap with the foreground pixels, while the others are set to 0. Erosion can be used to eliminate small, isolated foreground regions, separate connected objects, and smooth object boundaries.

Dilation: Dilation is the opposite of erosion and expands the boundaries of the foreground regions in a binary image. It is also performed using a structuring element, which slides over the image. A pixel in the dilated image is set to the maximum value (1) if at least one pixel of the structuring element overlaps with the foreground pixels in the original image. Dilation can be used to fill small holes, connect nearby objects, and smooth object boundaries. Another utilization of the dilation operation is to create a mask that is bigger than the segmented area but still follows the structure of the segmentation.

2.7.2 Compound Morphological Operations

Opening: Opening is a compound operation that consists of erosion followed by dilation using the same structuring element. It is useful for removing small foreground structures, such as noise or artifacts, while preserving the overall shape and size of the larger foreground regions. Opening can also be used to separate objects that are connected by thin bridges.

Closing: Closing is another compound operation, which involves dilation followed by erosion using the same structuring element. It is effective in filling small holes or gaps in the foreground regions while maintaining the overall shape and size of the larger structures. Closing can also be used to merge objects that are close to each other but not connected.

2.8 Related Work

In this section, a selection of relevant work on coronary artery segmentation and FFR estimation is presented.

2.8.1 ImageCAS: A Large-Scale Dataset and Benchmark for Coronary Artery Segmentation based on Computed Tomography Angiography Images

In 2022 a group of researchers compiled a comprehensive review of the current state of DL-based methods for segmentation of coronary arteries. They concluded that although the research on the subject has been plentiful and has shown great promise; Due to using proprietary datasets or unpublished code, comparison with other methods is impossible. In addition to these shortcomings, the datasets used by other researchers had very few cases, leading to suboptimal conditions for model training.

The authors proposed a large-scale dataset consisting of more than 1000 cases, from realistic clinical cases at the Guangdong Provincial People's Hospital from April 2012 to December 2018 [9]. This dataset is open to the public and is considerably larger than existing datasets. The authors compel new research to use this dataset as a benchmark in order to compare different network architectures, due to the large scale of the dataset. The scale of this dataset should help mitigate the lack of training data used in most previous work [8, 9], which should make most of the DL architectures perform closer to their full potential.

A baseline method was proposed in order for other researchers to be able to compare their work. This baseline method uses a combination of patch segmentation

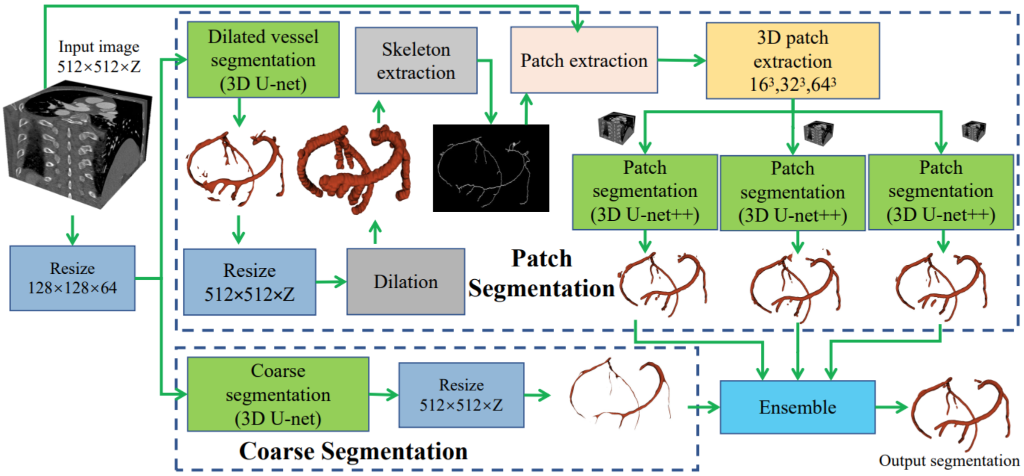


Fig. 3. Framework of the proposed baseline method.

Figure 2.12: Patch-based baseline method for ImageCAS

Source: [9]

and a coarse segmentation. The reason for using this combination is a tradeoff due to the memory limitation that made it impossible for the authors to do a direct segmentation on the whole volume at full resolution. The course 3D Unet model uses resized volumes with size 128x128x64 which gives a rough outline of the coronary arteries, with the full spatial cohesion intact. This model leads to a segmentation that has good spatial awareness but lacks the fine details of the coronary arteries. The patch segmentation part uses a preliminary step in order to select appropriate patches along the arteries. This preliminary step consists of a 3D Unet model that has a modified loss function biased to predict a segmentation that is larger than the original. This segmentation is then dilated further using morphology in order to ensure that the predictions are connected along the arteries. The dilated prediction is then skeletonized, then everything else than the two largest bodies is discarded, due to the domain knowledge that there is only two main coronary arteries. The skeleton is used as the basis for extracting full-resolution patches of size 16^3, 32^3 and 64^3, having the skeleton as the center point. These patches are then trained with their corresponding 3D Unet ++ [12] model and serve as a fine prediction of the arteries, due to the full-resolution input. The patch segmentation for each patch size is then combined with the coarse segmentation as an ensemble, see Figure 2.12.

With the proposed baseline method the dice score on the test set was 0.8296 with dilation and 0.8221 without the dilation step. In their ablation study, they also showed that the model performed best with the 64^3 patch size, although the ensemble led to the highest score when combined with the dilation step.

2.8.2 Automated segmentation of normal and diseased coronary arteries – The ASOCA challenge

In 2020 the ASOCA challenge invited contestants to perform semantic segmentation on 60 cases of CTCA images from the Coronary Atlas [8, 45]. The 60 cases were selected such that the division of patients with and without reported coronary disease was equally represented. 20 of the 60 annotations were kept hidden from the contestants and were used as the test set to evaluate the model predictions.

All the top submissions to the challenge were variations of the CNN U-Net architecture, including 2D Unet [36], 3D Unet [13] and nnU-Net [11]. The winning submission with a dice score of 0.87 was achieved by combining nnU-Net together with a Scale map generation [46].

Pre-processing filters, like the Frangi vesselness filter [47], was used to improve contrast and suppress other organs. Post-processing methods based on connected components have helped improve model performance by removing small disconnected components. Soft Dice Loss was the most common objective function, often combined with cross entropy or focal loss to handle class imbalance. Notably, healthy vessels were easier to segment than diseased ones, and annotator variability seemed to be higher on healthy vessels. Most methods had high precision but lower recall values, with segmentation becoming less reliable towards distal sections of the arteries.

2.8.3 Vascular Modeling Toolkit (VMTK)

The Vascular Modeling Toolkit (VMTK) is an open-source, extensible library and collection of tools developed to facilitate the processing and analysis of vascular structures in medical imaging data. VMTK is designed to support researchers and clinicians in developing novel computational models and simulations for the study of vascular anatomy, blood flow, and other physiological processes [48].

One of the primary applications of VMTK is the extraction and analysis of vascular structures from medical imaging data, particularly from Computed Tomography Angiography (CTA) and Magnetic Resonance Angiography (MRA) scans. The toolkit supports a range of segmentation techniques, including level-set methods and region-growing algorithms, to delineate the lumen and vessel walls. Additionally, VMTK provides tools for centerline computation and surface reconstruction, which can be used for subsequent geometric analysis and computational fluid dynamics simulations.

VMTK Centerline Extraction

In essence, centerlines are conceptualized as the shortest paths traced between two extreme points, weighted accordingly. These paths are not randomly dispersed in space; they are constrained to navigate the Voronoi diagram of the vessel model. The Voronoi diagram is a geometric construct representing the locus of centers of maximal inscribed spheres within the vessel. A sphere is considered maximal when it cannot be contained within any other inscribed sphere [49].

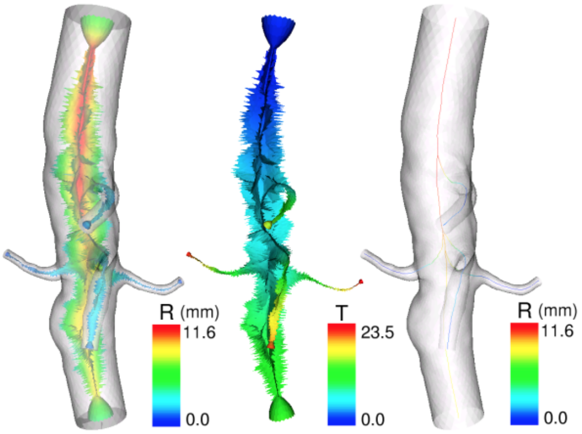


Figure 2.13: Centerline extraction. Left: aorta model and its embedded Voronoi diagram, R represents the radius of the maximum inscribed sphere. Middle: solution of the Eikonal equation from the inlet (from top). Right: centerlines backtraced from the outlets to the inlet

Source: [49]

Centerlines are defined as paths on the Voronoi diagram sheets that minimize the integral of the maximal inscribed sphere radius along the path, which is the same as identifying the shortest paths using the radius as the metric. This is achieved by propagating a wave from a source point, using the inverse of the radius as the wave speed, recording the wave arrival time on all points of the Voronoi diagram, and backtracing the line from a target point along the gradient of arrival times. This propagation is described by the Eikonal equation. As centerlines are defined on Voronoi diagrams, each point of the centerline corresponds to a maximal inscribed sphere radius. See Figure 2.13 for a visual representation of this process.

2.8.4 Machine learning augmented reduced-order models for FFR-prediction

As discussed in Section 2.1, FFR can be measured using a pressure sensor via a catheter. This is the most reliable way of measuring the FFR but it also has some substantial risks and costs attached to it.

In order to reduce the need for invasive procedures in the CAD diagnosis phase, FFR can be estimated by analyzing a coronary artery segmentation. In a recent study, the authors explored the incorporation of physics-based knowledge into machine learning models for prediction of FFR [17]. In this study, ANN are trained to predict pressure losses in coronary arteries using data obtained from solving the incompressible Navier-Stokes (iNS) equations. The coronary flow and geometrical data are used as inputs to train a purely data-driven ANN.

The authors investigated two methods for incorporating prior physics-based knowledge from a reduced-order model (ROM) into ANNs that predict pressure losses across stenotic and healthy coronary segments. The first method involves training an ANN to predict the discrepancy between the ROM and (iNS) pressure loss. The second method augments the data by including the ROM pressure loss prediction as an input feature to an ANN that predicts pressure.

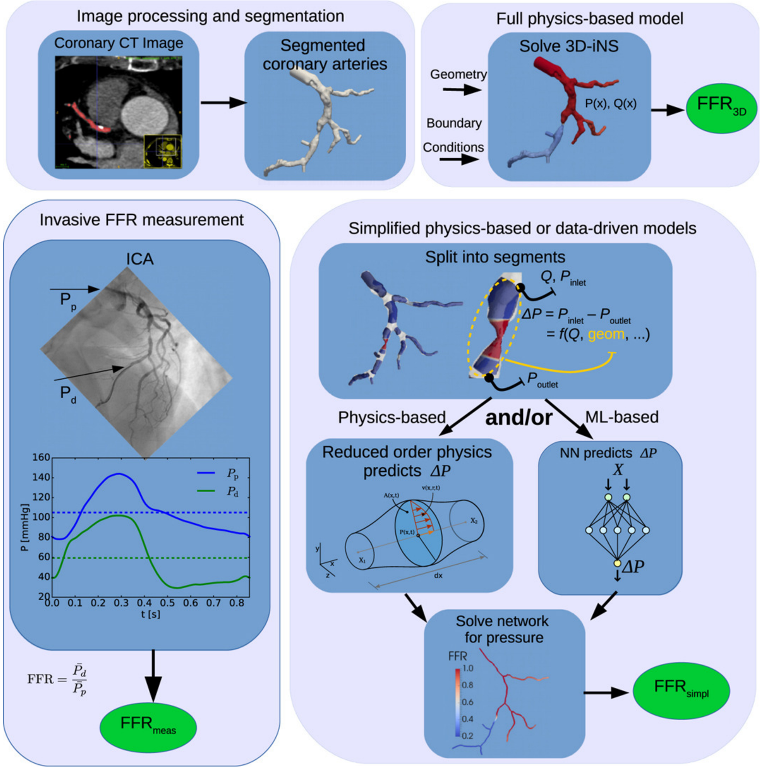


Figure 2.14: FFR estimation illustration

Source: [17]

Both approaches for incorporating prior knowledge from the ROM significantly improve the prediction of pressure losses across healthy and stenotic segments compared to the purely data-driven approach, particularly when there is a limited amount of data. By incorporating NN predictions of coronary segment pressure losses into a coronary network model, the study achieves FFR predictions with an error standard deviation of 0.021 with respect to the calculated FFR from solving the iNS equations. This performance is comparable to the standard deviation of repeated FFR measurements, which is 0.018 [50].

Figure 2.14 outlines the different proposed ways of acquiring FFR, in addition to the normal invasive method. Note that 3D INS, ROM and ANN approach uses a segmentation model of the coronary arteries as the input for the calculation. In order to utilize the 3D segmentation for the FFR prediction, it has to undergo some preliminary pre-processing. One of these pre-processing steps are to localize the centerline as well as splitting the segmentation into different segments. Both of these pre-processing steps are performed using VMTK.

Fully automatic FFR prediction pipeline

During conversations with Fredrik E. Fossan, the primary author of the FFR estimation paper [17], the possibility of creating a fully automated pipeline from CTCA scans to predict FFR was discussed. Fossan pointed out that while a high-quality automatic segmentation of coronary arteries is essential, the centerline extraction tool from VMTK requires some user interaction to be executed. This interaction involves placing the inlet and outlet positions in relation to the segmentation model. If these two steps could be automated, it would enable the implementation of a fully automated pipeline utilizing the proposed FFR estimation technique.

Chapter 3

Methodology

In this chapter, the methodology for the experiments performed in this thesis will be explained, as well as a section describing the datasets on which the experiments were performed on. In addition to the experiment methodology, the tools used to perform the experiments, as well as analysis are also explained.

3.1 Tools

3.1.1 PyTorch

PyTorch is a widely used open-source framework for working with deep learning in Python. Parts of the framework are written in C++ and CUDA for efficiency and then wrapped in Python for quick development [51]. The framework also has a graph-based automatic differentiation package, which makes backpropagation easy to work with.

3.1.2 PyTorch Lightning

PyTorch Lightning is a high-level wrapper for PyTorch. It is designed to simplify the process of training, validating, and testing deep learning models while maintaining the flexibility and expressiveness of PyTorch [52]. PyTorch Lightning provides a structured approach to organizing deep learning code and reduces boilerplate code by abstracting away much of the training loop and handling of devices, such as CPUs and GPUs.

The central abstraction in PyTorch Lightning is the LightningModule, which encapsulates the neural network architecture, loss functions, and optimization algorithms. This module is designed to be easily extensible, allowing users to define custom training and validation steps, as well as configure data loading and distributed training settings. PyTorch Lightning also includes a range of built-in utilities for logging, checkpointing, and visualization, making it easier to track and manage model training and evaluation.

3.1.3 MONAI

Medical Open Network for AI (MONAI) is an open-source, PyTorch-based framework specifically designed for deep learning in medical imaging [53]. MONAI aims to provide a comprehensive set of tools and functionalities for the development, training, and evaluation of medical imaging models while maintaining high performance, flexibility, and extensibility.

One of the main strengths of MONAI is its domain-specific set of modules and components tailored for medical imaging tasks. These include data loaders and transforms for handling various medical image formats, specialized layers and architectures for medical image analysis, and domain-specific loss functions and evaluation metrics. By providing these components, MONAI simplifies the development process and allows researchers and practitioners to focus on their specific tasks rather than implementing common functionalities from scratch.

Another important aspect of MONAI is its active community and ongoing development. The MONAI community consists of researchers, developers, and clinicians who contribute to the project by sharing their expertise, providing feedback, and implementing new features. This collaborative environment helps MONAI to continuously evolve and adapt to the needs of the medical imaging community.

MONAI is divided into three main components:

1. MONAI Core: This is the foundational library of MONAI, providing the main functionalities for medical imaging research. It includes a set of PyTorch-based tools for healthcare imaging tasks such as data pre-processing and augmentation, defining complex network architectures, and various training strategies. The MONAI Core is flexible and interoperable, supporting numerous medical imaging-specific formats and tasks.
2. MONAI Deploy: This is the component of MONAI that enables the translation of models developed using MONAI Core (or other tools) into real-world clinical or research deployments. It provides a set of tools for packaging AI models and their associated workflows into deployable units. MONAI Deploy supports the deployment of these models across various healthcare IT environments, accommodating different deployment strategies such as local execution, server-client architecture, or even cloud-based execution.
3. MONAI Label: This is an intelligent open-source tool for the fast annotation of medical imaging datasets. It supports various interactive annotation operations and integrates AI-assisted annotation capabilities powered by models trained using MONAI Core. The aim of MONAI Label is to speed up

> the often tedious process of data annotation in medical imaging research, thus facilitating the development of larger, high-quality datasets for training AI models. MONAI Label has plugins in order to directly interact with multiple anatomical 3D viewing software like 3D Slicer, making the interaction between annotation, inference and model training easy to work with [54].

Monai Core was used extensively in this thesis.

3.1.4 3D Slicer

3D Slicer is an open-source software platform for medical image processing and three-dimensional visualization. Developed by the Slicer community, it offers a wide range of tools for segmentation, registration, and quantitative analysis of medical images [55]. Its modular architecture enables researchers and clinicians to easily extend its functionality through custom-built plugins, making it a popular choice for both research and clinical applications [56].

One of the key strengths of 3D Slicer is its support for various image formats, including Digital Imaging and Communications in Medicine (DICOM), Neuroimaging Informatics Technology Initiative (NIfTI) and many others, allowing seamless integration with existing medical imaging workflows. Additionally, 3D Slicer provides several built-in tools for manual and semi-automatic segmentation, as well as support for importing and exporting segmentations to and from other software packages [57].

3D Slicer has been employed in a wide range of medical imaging applications, including neurosurgery planning, radiotherapy, and image-guided interventions [55]. Its open-source nature, active development community, and extensive feature set make it a valuable tool for medical image processing and analysis.

3.2 Datasets

Three different datasets were used in this thesis, where the first dataset is a non-public dataset from St. Olavs Hospital in Trondheim Norway. The two other datasets are the datasets used in previous work as presented in Section 2.8.

St. Olavs Hospital Dataset comprises 117 prospectively enrolled outpatients from October 2018 to March 2021. These patients presented stable chest pain, low to intermediate pretest likelihood of CAD, and positive coronary CCTA, which led to their referral for ICA. Several exclusion criteria were applied, such as previous coronary revascularization, age over 75, BMI over 40, and specific medical conditions. CCTA images were acquired at St. Olavs University Hospital and five collaborating local hospitals, following current guidelines. The left and right coronary vessels' semantic segmentations were generated semi-automatically using Mimics software, with manual corrections and quality control by experienced radiologists.

Table 3.1: Stenosis severity categories

Category	Stenosis severity	FFR
1	Uncertain/unknown grade	
2	Low grade	FFR > 0.9
3	Intermediate	0.7 < FFR <= 0.9
4	Severe	FFR <= 0.7

This dataset also contains segmentations of clinically evaluated stenosis areas, as well as a classification of the severity of the stenosis. The categories for stenosis fall into four categories as seen in Table 3.1. For additional clinical characteristics for the dataset, see Table A.1

Coronary Atlas dataset features 60 CCTA cases obtained from the Coronary Atlas[45]. The selected patients were divided based on available medical reports, resulting in 30 patients with reported coronary disease and 30 without. Images were captured using a GE LightSpeed 64-slice CT scanner, employing retrospective ECG-gated acquisition at the late diastole time point for reconstruction. The resulting images exhibit anisotropic resolution, with an in-plane resolution of 0.3-0.4mm and an out-of-plane resolution of 0.625mm.

ImageCAS dataset consists of 3D CCTA images from 1,000 patients, captured using a Siemens 128-slice dual-source scanner. The dataset includes patients who underwent early revascularization (within 90 days) after being diagnosed with coronary artery disease. The data was collected at the Guangdong Provincial People's Hospital between April 2012 and December 2018 and comprised 414 female and 586 male patients. For each image, the left and right coronary arteries were independently labeled by two radiologists. In cases of discrepancy, a third radiologist intervened, and the final result was determined by consensus.

3.2.1 Dataset split

As the three datasets had some differences, the training, validation and test split was slightly different for each dataset.

For the St. Olavs Hospital and ImageCAS dataset, the split followed 70% for training, 20% for validation and 10% for testing. For the St. Olavs Hospital dataset, the additional information on stenosis severity guided the split such that there would be an equal proportion of each of the categories, except for category 1 (only 1 sample in this category) (Table 3.1). Because the Asoca Challenge had a hidden test set of 20 samples where the labels were only available through online submission, these 20 samples became the test set, where the ratio was 1:1 between

Table 3.2: Dataset Statistics

description	Coronary Atlas	St Olav	ImageCAS	Combined
Samples	60	117	1026	1203
Shape min	[512 512 168]	[512 512 200]	[512 512 166]	[512 512 166]
Shape max	[512 512 224]	[513 513 521]	[512 512 277]	[513 513 521]
Shape mean	[512 512 213]	[512 512 345]	[512 512 257]	[512 512 265]
Spacing min	[0.33 0.33 0.62]	[0.31 0.31 0.3]	[0.29 0.29 0.5]	[0.29 0.29 0.3]
Spacing max	[0.49 0.49 0.62]	[0.52 0.52 0.63]	[0.46 0.46 0.5]	[0.52 0.52 0.63]
Spacing mean	[0.41 0.41 0.62]	[0.42 0.42 0.4]	[0.35 0.35 0.5]	[0.36 0.36 0.49]
Label intensity min	-370	-199	-1015	-1015
Label intensity max	1843	1411	3069	3069
Label intensity .5 percentile	147	115	-183	-137
Label intensity 99.5 percentile	653	678	603	613
Foreground percentage min	0.02	0.02	0.05	0.02
Foreground percentage max	0.1	0.12	0.36	0.36
Foreground percentage mean	0.05	0.06	0.17	0.15

diseased and healthy patients. The remaining 40 samples were split 80% for training and 20% for validation, both with equal distribution of healthy to diseased.

3.3 Model implementation and validation

In this section, the implementation details for each of the models trained will be explained. As well as the methodology for validating the accuracy of the different models.

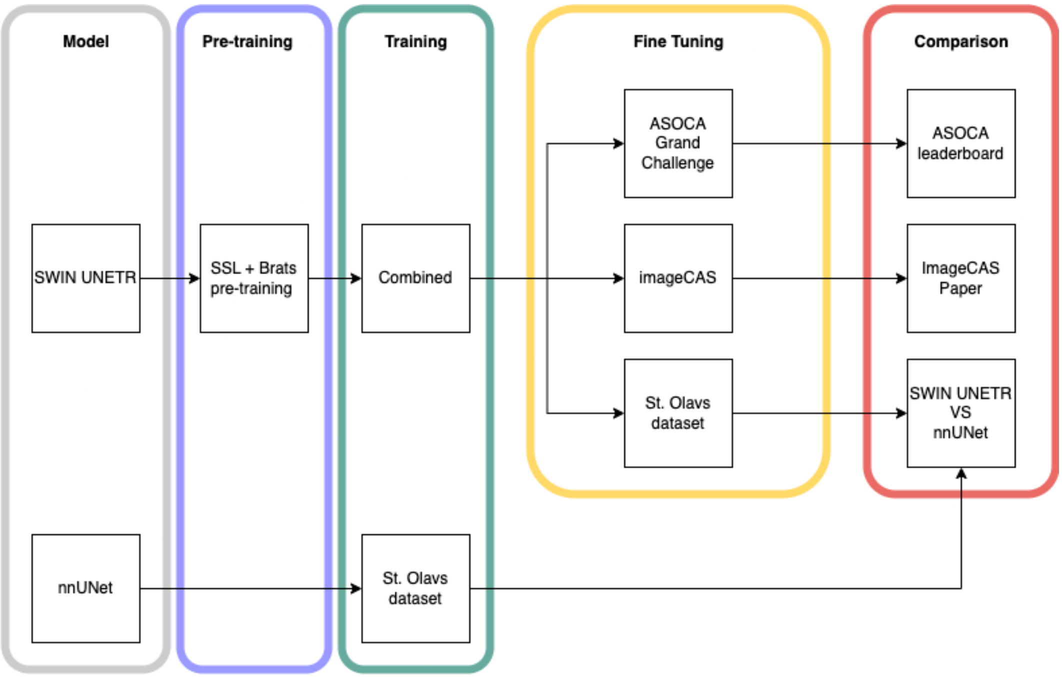


Figure 3.1: Experimental overview

The experimental overview is divided into model selection, pre-training, training, fine-tuning and finally the validation of the results, see Figure 3.1. The comparison part refers to each of the three datasets presented in Section 3.2, and combined refers to the combination of all three datasets. The idea behind training a combined model is to increase the number of samples as all three datasets are in the same domain. The fine-tuning step is to make sure the model is tuned to the specific image quality, spacing and consistency of the segmentation style. The segmentation style is mostly referring to where the radiologists set the distal cut off, but also general variability between radiologists and imaging quality. The fine-tuning step should make the final Dice Score (DSC) higher given there are differences in these areas.

3.3.1 SWIN UNETR

The transformer model implemented in this thesis is SWIN UNETR, the reason for the selection of SWIN UNETR was the impressive real-world results on 3D segmentation of medical images [15, 43, 58]. In this section, the model implementation and pipeline are explained for the SWIN UNETR model.

Pre-trained weights

In Section 2.5.3 an SSL pre-training method for SWIN UNETR was introduced. In the research paper outlining this method, they provide the pre-trained model weights, for people to download. This model was SSL pre-trained using 5050 3D CT samples from 5 public medical datasets [43]. As mentioned in Section 2.3.5 SSL techniques do not use labeled targets so the segmentations provided in these datasets are irrelevant. The raw CT scans include the head, neck, lungs and colon. Due to the nature of CT scans the images would also contain some portions of the surrounding anatomical structures. After this SSL pre-training the model was then fine-tuned on the Beyond The Cranial Vault (BTCV) dataset and is currently 1st place on papers with code [58, 59] for the BTCV multiclass segmentation task.

To adapt the pre-trained weights from the BTCV dataset for coronary artery segmentation, the output layer was modified to a single channel output, replacing the original 13-channel output. As none of the 13 organs in the BTCV dataset feature coronary arteries, the benefits of using these pre-trained weights stem from the broad modeling capabilities of the Self-Supervised Learning (SSL) strategy. This practice, coupled with transfer learning from a slightly dissimilar domain, is a conventional approach in transfer learning applications.

Pre-processing

All of the SWIN UNETR models followed the same pre-processing pipeline implemented with MONAI transforms. Although the same pipeline was implemented for all models, some of the hyper-parameters for the baseline model and the fine-tuned model were adjusted on the basis of the specific dataset statistics as shown in table Table 3.2.

The pipeline consists of the following steps:

Resample voxel spacing: The voxel spacing is the physical distance between each voxel and is usually defined in mm. The spacing is a product of the imaging hardware and the settings used for each imaging session. Due to the variance of voxel spacing in the samples, a resampling method that uses the mean voxel spacing of the given dataset was chosen. This makes sure that the physical distance between each voxel is consistent between every sample. In effect, this transformation will make images with a lower spacing than the mean larger in terms of voxels and the opposite for images with larger than mean spacing, in each respective dimension.

Scaling intensity range: The HU intensity was clipped at -200 to 1411 in order to remove some of the complexity of the surrounding areas around the coronary arteries. The range was set by selecting the min and max intensity, considering only voxels that are segmented as coronary arteries. The specific values were selected by using the values from the St. Olavs hospital dataset as this dataset had a smaller range but still encapsulated the .5 and 99.5 percentile label intensity of all three datasets. The reason for not choosing the combined .5 percentile and 99.5 percentile as the threshold was to make sure sufficient information was available in order to perform the segmentation. In addition to the clipping, this range was then normalized to values between 0 and 1 in order for the network to use the values more efficiently.

Cropping foreground: In order to further remove unnecessary information, each image in the training set was cropped to the smallest bounding box that would fit all labeled voxels for the current image. In effect, this transformation reduces the overall impact of the class imbalance, by reducing the background to foreground ratio.

Patch extraction: In order to adhere to the memory limitations while training, 3D patches of size 160^3 was selected. The selection criterion was that each patch had a 75% chance of having a coronary artery label as the center voxel. The oversampling of positive labels was performed to further reduce the class imbalance.

Post-processing

In order to reduce the number of artifacts in the predicted segmentation, all connected segmentations with a voxel volume of less than 1000 voxels were removed.

Validation step under training

As the models were trained on 160^3 patches, the validation step utilized a sliding window inference with an overlap of 25%. This method works by predicting 160^3 patches with a sliding window and the results are combined into a full volume.

Preliminary experiment for assessing hyper-parameter setup

As discussed in Section 2.3.5, there is a variety of loss functions to choose from when determining the training strategy. The primary objective of this selection was to maximize the Dice Similarity Coefficient (DSC) by minimizing the impact of class imbalance (Section 2.6), which is naturally present in the datasets. The SWIN UNETR model was trained on the St. Olav Hospital dataset for 500 epochs. In addition to evaluating loss functions, the proportion of samples with a positive label at the center was tested, with 50% and 75% probability. The final preliminary test aimed to investigate whether pre-trained weights from [59] would have a positive impact on the training process. Although pre-trained weights have been demonstrated to yield better results in previous studies, the benefits may be limited if the domains of the datasets are not highly similar.

Base SWIN UNETR model

The setup for training the base model consisted of selecting the best hyper-parameters for training as concluded in the preliminary study. The full hyper-parameter setup for the training is shown in table Table 3.3. Note that the batch size of 1 refers to 1 patch from each sample in each epoch, which is the case for all the SWIN UNETR models trained in this thesis.

Table 3.3: Base model hyper-parameters

Parameter	Value
Intensity max	614
Intensity min	-138
Spacing dim	[0.37 0.37 0.5]
Patch size	[160 160 160]
Learning rate	0.0002
Linear warmup epochs	5
Epochs	500
Batch Size	1
Loss function	Dice + Focal loss
Pretrained	BTCV + SSL [59]
Feature Size	48

The combined dataset used to train this model was simply combining the existing subsets of the training, validation and test split for each dataset as explained in Section 3.2.1.

Fine tuned SWIN UNETR models

After the preliminary base model was trained, an additional fine-tuning step was performed on each of the three datasets. The goal of this process was to optimize the model on the specific annotation style, imaging quality and intensity levels present in each dataset. Each model was trained for 500 epochs. The model hyper-parameters for each of the fine-tuned models were mostly the same as the base model but the max and min intensity was set to -200 to 1411 instead of 0.05 and 0.95 percentile as in the base model due to experimental results. The linear warmup was increased to 50 epochs for St. Olavs Hospital dataset and Coronary Atlas due to the number of samples and the spacing dims was set to the mean value for each respective dataset, see Table 3.4.

Code implementation

The code for the training loop, pre and post-processing as well as data handling was done using MONAI in conjunction with PyTorch Lightning. The actual model implementation was directly used from the MONAI library, and the whole coding

Table 3.4: Fine-tuned model hyper-parameters

Parameter	St. Olavs Hospital	Coronary Atlas	ImageCAS
Intensity max	1411	1411	1411
Intensity min	-200	-200	-200
Spacing dim	[0.42 0.42 0.4]	[0.41 0.41 0.62]	[0.35 0.35 0.5]
Patch size	[160 160 160]	[160 160 160]	[160 160 160]
Learning rate	0.0002	0.0002	0.0002
Linear warmup epochs	50	50	5
Epochs	500	500	500
Batch Size	1	1	1
Loss function	Dice + Focal loss	Dice + Focal loss	Dice + Focal loss
Pretrained	Base model	Base model	Base model
Feature Size	48	48	48

project was inspired by the official implementation for the SWIN UNETR BTCV repository [59]. The code was then rewritten in the PyTorch Lightning framework, in order to make use of distributed training and data handling in a cleaner way than the original implementation.

3.3.2 nnU-Net

In addition to SWIN UNETR, a nnU-Net model was trained on the proprietary St. Olavs Hospital. The reason for training this model was to have a strong baseline for validating the SWIN UNETR model accuracy against, as there is no other public baseline on this dataset. The nnU-Net model was trained using stock settings as this is one of the main features of the nnU-Net, namely that it is self-configuring to the dataset. The training was done for 1000 epochs without any additional pre-training. The code used to run the model training was gathered from the official github repository by the authors [60], and was only modified in order to accommodate the St. Olav Hospital dataset.

3.3.3 Comparison of the test set scores

For the SWIN UNETR model, the Dice Score (DSC) on the test set was calculated using a sliding window inference with 25% overlap and with a Gaussian distribution of the blending weights. During the training, the validation was done using images normalized to the average voxel spacing, but for the testing, it was changed to the original resolution, due to a significant accuracy boost that was observed while testing different inference strategies.

In order to compare the DSC to previous work three different strategies were followed:

- The ImageCAS dataset was evaluated on the test set and compared with the author's baseline method described in section Section 2.8.1.
- The Coronary Atlas dataset was validated by submitting our results to the ASOCA Challenge.
- The St. Olavs Hospital dataset was validated by comparing the DSC and 95th Hausdorff distance between nnU-Net and SWIN UNETR model predictions. Both metrics are calculated using the same Monai code for consistent calculations. Additionally, the prediction of the two models was also compared qualitatively by visualizing the predictions in 3D Slicer.

3.3.4 Compute

Transformer models are known to require a lot of GPU memory, this is either due to the model size or the size of the input. In our experiments, the biggest limiting factor is the size of the medical images, as the GPU needs to be able to hold the activations in memory during the forward pass, as well as the gradient for the backpropagation. As the complexity of the network expands, the gradients and the activations also increase, and the transformer models are typically more complex than their CNN-based counterparts. The GPU memory need for our SWIN UNETR models required a minimum of 20GB; Given the hyper-parameters chosen for model training, with a patch size of 160^3 and batch size of 1. For the nnU-Net model the automatic hyper-parameter tuning is dynamically aimed to hit 16GB of memory usage, either by controlling the patch or the batch size.

In order to meet the need for a minimum of 20GB for the SWIN UNETR model training, a dual NVIDIA V100 with 32GB VRAM system was utilized. For the nnU-Net model training a single NVIDIA A40 with 16GB was utilized.

A bottleneck for the model training speed is the large file size of 70-100MB for each sample. Due to the large size, most of the training time will be used for reading from disk when using an ordinary dataset object from MONAI. In order to speed up this process, dataset caching of samples along with the deterministic transforms in the pre-processing step. MONAI CacheDataset was used for this process and greatly improved the training speed. In order to fully utilize this strategy the system has to have sufficient RAM, which for the combined dataset amounted to 600GB of RAM.

3.4 Post segmentation analysis

In addition to performing the 3D segmentation of the coronary arteries, various post-segmentation analysis was performed. This post-segmentation analysis included centerline extraction, FFR estimation and plaque localization.

3.4.1 Plaque localization

Although plaque in CCTA images is associated with a wide range of HU intensities, a portion of the calcified plaque can be identified as having HU values above the HU intensity range from the segmented coronary arteries. Looking at the dataset statistics from Table 3.2, the highest 99.5 percentile upper bound for the coronary artery voxel intensities was observed at 678HU. This upper bound indicates that most values above this range should not be a part of the luminal voxel intensity. Setting a threshold of 850HU should ensure that the area is most likely associated with calcified plaque, given that it is located in close proximity to the segmentation.

Using the information about the typical HU intensity range of calcified plaque and the information from the dataset statistics, it is possible to perform a segmentation on this type of plaque. The first step in this process is to use the morphological operation dilation in order to expand the size of the predicted segmentation. The second step is to make a logical masked area given by the original image $>$ 850HU. The plaque could then be segmented using the product of the dilated segmentation and the logical threshold mask.

This plaque segmentation is useful for the visual analysis of the predicted segmentation. Additionally, this information might be useful for radiologists as it gives a rough estimation of calcified plaque location. This rough estimation can be used to guide further investigation in the raw CCTA data.

3.4.2 Centerline Extraction

In order to perform FFR estimation using the technique outlined in Section 2.8.4, it is necessary to get the centerline model as well as inlet and endpoint positions. The centerline model was extracted using the VMTK's extract centerline function as explained in Section 2.8.3. The VMTK standalone package does not have the ability to automatically extract the inlet and endpoints of the coronary arteries, this was solved using the SlicerVMTK extension for 3D Slicer, and scripted in Python with the 3D Slicer Python environment. This technique allows for a fully automatic centerline extraction, given that the prediction was split into the largest and second largest segment from the prediction and performed on each segment separately.

3.5 FFR Estimation

As explained in Section 2.8.4, the proposed physics informed ANN method from previous work for FFR estimation requires a 3D segmentation of the coronary arteries in order to be performed. In this experiment, the segmented arteries are automatically generated by using the model predictions from the fine-tuned SWIN UNETR model trained on the St. Olavs Hospital Dataset, with samples from the corresponding test set. The reason for using this model is that the St. Olavs Hospital dataset contains clinically evaluated stenosis location as well as corresponding physical FFR measurements, which can be used for comparing the estimated FFR against. As the previously proposed FFR estimation technique required manual intervention in order to acquire the centerline needed to perform the estimation, centerlines from our proposed automatic centerline extraction are used as input. Using this automatically generated centerline would give insight into the feasibility of automating the whole FFR estimation process.

The previously proposed trained ANN with ROM input features were then used to estimate FFR on each of the model predictions from the test set samples in the St. Olavs Hospital dataset. In addition to comparing these values to the physically measured FFR, FFR is also estimated using the ground truth segmentation, and a set of predictions where obvious prediction errors are manually corrected.

Chapter 4

Experiments and Results

This chapter unveils the findings derived from the experiments conducted for this thesis. The subsequent experiments were performed to adequately address the research questions:

Experiment 1: Hyper-parameter Selection
A preliminary experiment in order to select the best hyper-parameters for training the SWIN UNETR models.

Experiment 2: SWIN UNETR baseline model
All three datasets are combined in order to maximize the number of samples used for training the SWIN UNETR baseline model.

Experiment 3: Fine Tuning SWIN UNETR
Fine-tuning the SWIN UNETR models on the three datasets, and a comparison with previous work.

Experiment 4: nnU-Net vs SWIN UNETR
Training a nnU-Net model on St. Olavs Hospital dataset in order to compare to the results from SWIN UNETR model.

Experiment 5: Plaque Localization
Localizing plaque by utilizing morphology and HU thresholds.

Experiment 6: Automatic Centerline Extraction
Extracting centerline from model prediction.

Experiment 7: FFR Estimation
Using segmentations produced by SWIN UNETR and automatically produced centerlines in order to estimate FFR using previous work.

4.1 Experiment 1: Hyper-parameter Selection

In the search for the best set of hyper-parameters to train SWIN UNETR model, each new hyper-parameter was used to train the model on the St. Olavs. Hospital dataset for 500 epochs. If the new hyper-parameter gave a better DSC, the hyper-parameter was included in the next iteration.

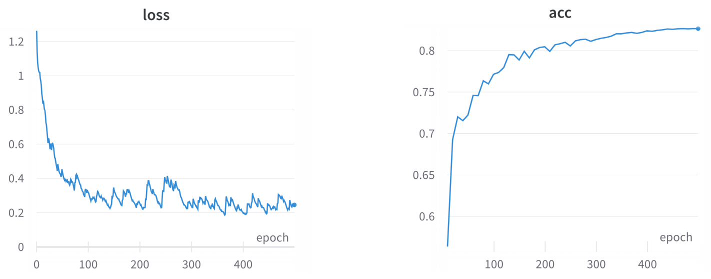
Variation	Loss Function	Positive label probability	Pretrained	DSC
1	Dice + Cross entropy	50%	No	0.7940
2	Dice + Focal	50%	No	0.8127
3	Dice + Focal	75%	No	0.8329
4	Dice + Focal	75%	Yes†	0.6824
5	Dice + Focal	75%	Yes	**0.8447**

Table 4.1: Hyper-parameter selection results, where † denotes that the encoder of the pre-trained weights was frozen through the training. Positive label probability refers to the probability of the randomly selected patch having a positively labeled voxel in the center.

The results of this initial study concluded that using Dice + Focal Loss, positive label probability of 75% and using pre-trained weights with unfrozen weights achieved the highest DSC, see Table 4.1. These hyper-parameters were therefore used to train the baseline model and the fine-tuned models.

4.2 Experiment 2: SWIN UNETR baseline model

The ImageCAS dataset consists of 1026 samples, which is considerably larger than the two other datasets with 60 and 117 samples. In order to take full advantage of this data all three datasets are combined in order to train a baseline SWIN UNETR model. Due to the number of samples in this combined dataset, the 500 epochs took about 4 days to complete with 2 NVIDIA V100 GPUs. The checkpoint was saved at the epoch with best DSC (0.8265).



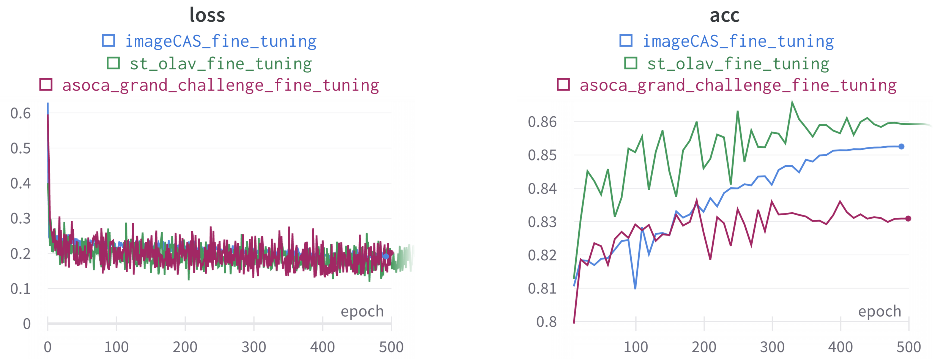
(a) Training loss

(b) Validation accuracy (DSC)

Figure 4.1: Training loss and validation accuracy during training of the baseline model

4.3 Experiment 3: Fine Tuning SWIN UNETR

Due to potential differences in the three datasets, a new SWIN UNETR model was fine-tuned for each of the datasets, using the baseline model from experiment 2 as the pre-trained weights. Although the accuracy and loss did converge while training the combined model (Figure 4.1), it is apparent that the fine-tuning step still had some impact, as all three models were able to further improve with respect to DSC and loss as seen in Figure 4.2.



(a) Training loss

(b) Validation accuracy (DSC)

Figure 4.2: Training loss and validation accuracy for fine-tuned SWIN UNETR

The final accuracy for each of the three fine-tuned models is summarized in Table 4.2. The results from the fine-tuning show that all models did benefit from the fine-tuning step with an increased DSC compared to the baseline model. While testing different inference strategies, it was observed that performing inference with original spacing, unlike the normalized spacing used in validation resulted, in a significantly higher DSC. This change accounted for about 80% of the accuracy gain between the validation and test score, while the remaining gain came from post-processing. Note that experiments were performed with original spacing for training, which resulted in an inferior DSC, unfortunately, this result got lost and is therefore not presented in this thesis.

Model	Validation DSC	Test DSC	Test Post processed DSC
imageCAS	0.8528	0.8595	0.8614
Coronary Atlas	0.8382	NA	0.8663 †
St Olavs Hospital	0.8657	0.8808	0.8855

Table 4.2: Accuracy results for fine-tuned SWIN UNETR models. † denotes that the test DSC was evaluated through ASOCA grand challenge website.

4.3.1 Comparison with previous work

In order to evaluate the model performance, the results are compared with results from previous work as described in Section 3.3.3.

The fine-tuned model trained on the Coronary Atlas dataset was used to perform segmentation on the 20 samples in the test set. The predicted segmentations were then post-processed and submitted to ASOCA challange by online submission. In the online leaderboard, this submission achieved 7th place with a mean DSC equal to 0.8663, which is about 0.02 less than the top submission, see Table B.1 [61]. The results are also compared with the ASOCA challenge paper results [8], where all submissions are rounded to two decimal places, see Table 4.3.

Submission	DSC
Top 3 Ensemble	**0.88 ± 0.04** †
Top 5 Ensemble	**0.88 ± 0.04** †
Top 7 Ensemble	0.87 ± 0.04
Submission 1	**0.87 ± 0.04**
Submission 2	0.84 ± 0.05
Submission 3	0.86 ± 0.07
Submission 4	**0.87 ± 0.05**
Submission 5	0.8 ± 0.04
Submission 6	0.84 ± 0.06
Submission 7	0.78 ± 0.1
Submission 8	0.73 ± 0.05
Our	**0.87 ± 0.05**

Table 4.3: Results from ASOCA Grand challenge paper. Bold indicates the best submission and † indicates the best ensemble submission.

Source: [8]

The model fine-tuned on the ImageCAS dataset was evaluated by comparing DSC from the post-processed test set predictions with the various models presented in the ImageCAS paper [9]. Table 4.4 shows the results from the fine-tuned model and the different methods proposed in the paper, where the fine-tuned SWIN UNETR model outperforms the best baseline by $\sim$ 0.03 in DSC.

Method	Input type	Input size	DSC
Direct segmentation (3D FCN)	Full Image	512 x 512 x 256	0.8058
Patch segmentation (3D U-net)	Patch	64 x 64 x 64	0.7201
Tree data based segmentation (3D TreeConvGRU)	Tree	N x 16 x 16 x 4	0.6878
Graph based segmentation (GCN)	Graph	N x 32	0.7061
Baseline method (3D U-net and Unet++)	Patch	128 x 128 x 64	0.8296
Our (SWIN UNETR)	Patch	160 x 160 x 160	**0.8614**

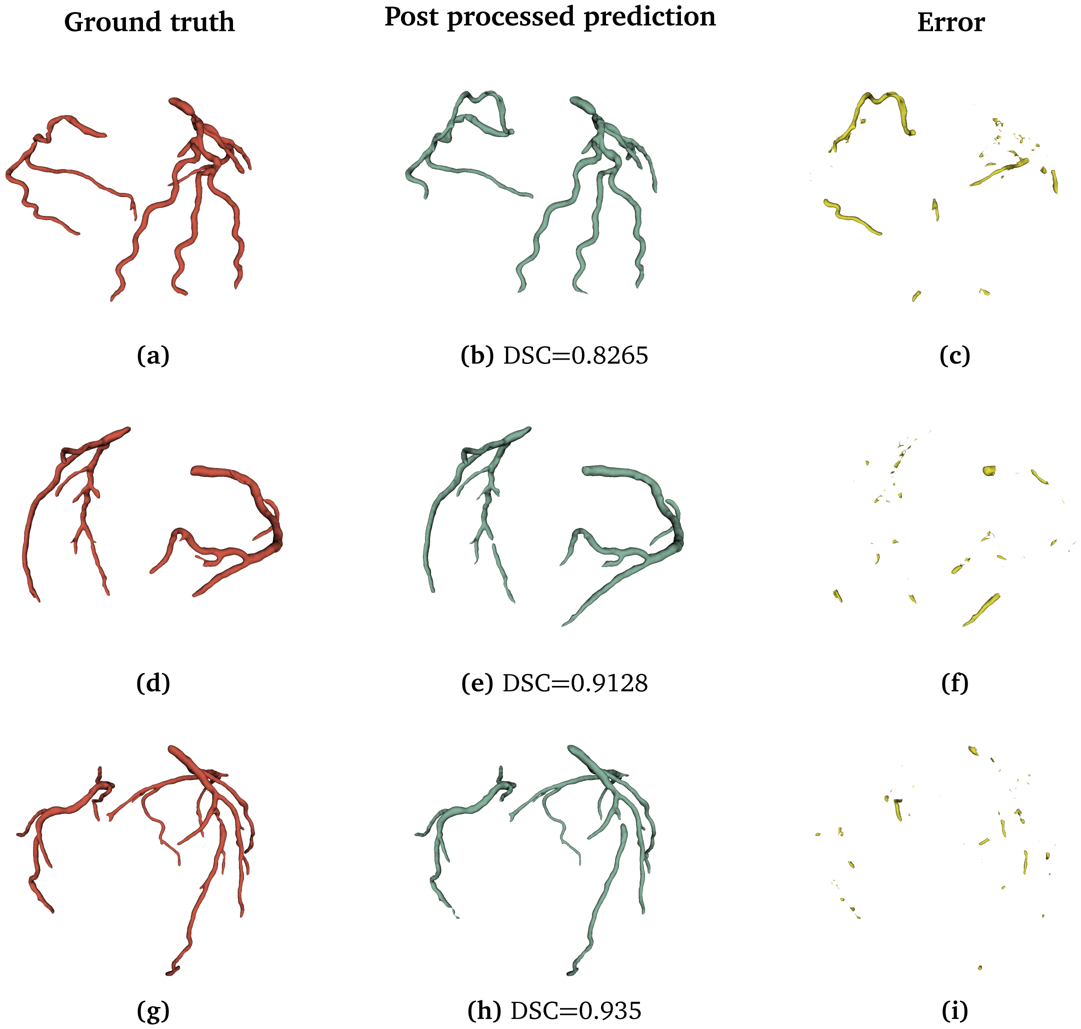
Table 4.4: ImageCAS results

Source: [9]

4.3.2 Qualitative results

In order to complement the quantitative results, a visual inspection of the model predictions was performed. In this section, the ground truth and post-processed prediction from the fine-tuned model trained on St. Olavs Hospital are compared in order to reveal the quality of the predictions and to highlight various faults in the segmentation.

Figure 4.3 shows the post-processed model prediction, ground truth and the total error (FP and FN), with the corresponding DSC under each prediction. Looking at the error and the corresponding DSC gives a visual indication of the amount of error a particular DSC is associated with.



(a) **(b)** DSC=0.8265 **(c)**

(d) **(e)** DSC=0.9128 **(f)**

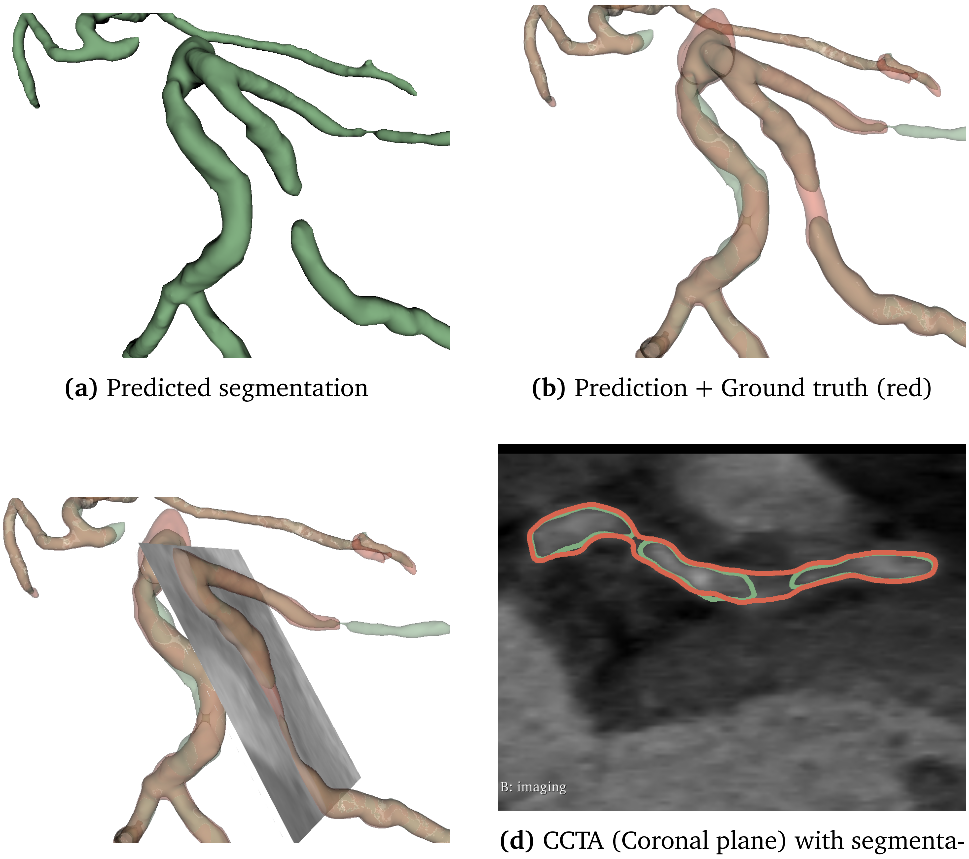
(g) **(h)** DSC=0.935 **(i)**

Figure 4.3: SWIN UNETR prediction error

False Positives and False Negatives

False negatives leading to a disjointed segment and false positives in the distal areas of the coronary arteries were observed in multiple of the predicted segmentations. In order to explore these issues, a visual inspection using 3D slicer was performed. The visual inspection compares the post-processed predicted segmentation as a 3D render and by inspecting relevant CCTA raw data in the problem areas. All examples in this section feature post-processed predictions from the SWIN UNETR model fine-tuned on the St. Olavs Hospital dataset.

In the first example, the prediction is disjointed in an area with no registered stenosis and no apparent narrowing. From the CCTA slice, the luminal intensity is slightly lower in the false negative areas than the true positives, see Figure 4.6.



(a) Predicted segmentation

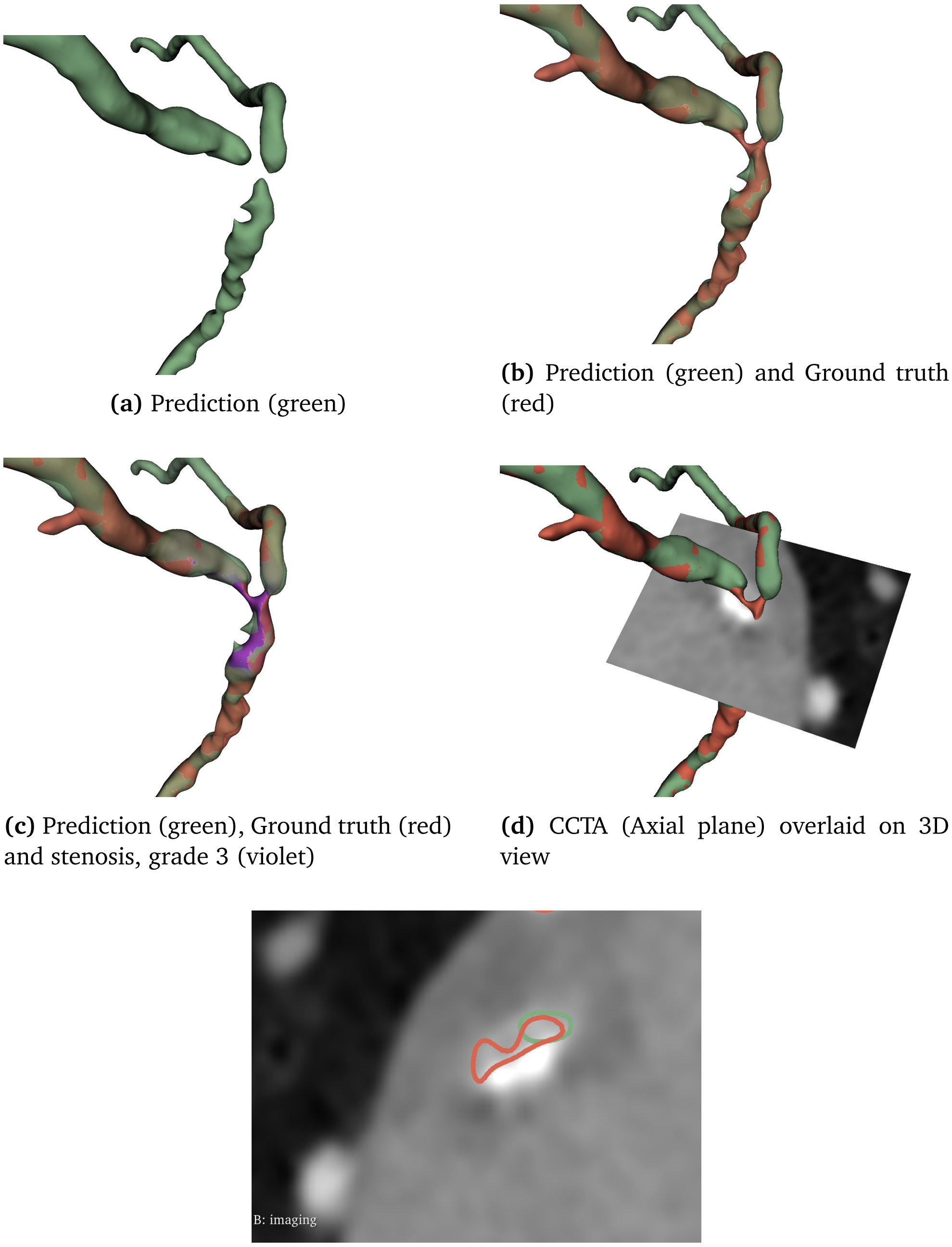
(b) Prediction + Ground truth (red)

(c) Prediction + Ground truth (red)

(d) CCTA (Coronal plane) with segmentation contours

Figure 4.4: The disjointed segmentation (green) is compared with ground truth (red) in both 3D view, and in the coronal plane of the raw CCTA data.

In the second example where false positives are leading to a disjointed segmentation, the area is associated with a stenosis of category 3. From the 3D render it is apparent that the ground truth is significantly narrowed in the problem area, see Figure 4.5b. The CCTA slice shows a high intensity in this area, which is associated with calcified plaque buildup. See link for a video demonstration of the region of interest.



(a) Prediction (green)

(b) Prediction (green) and Ground truth (red)

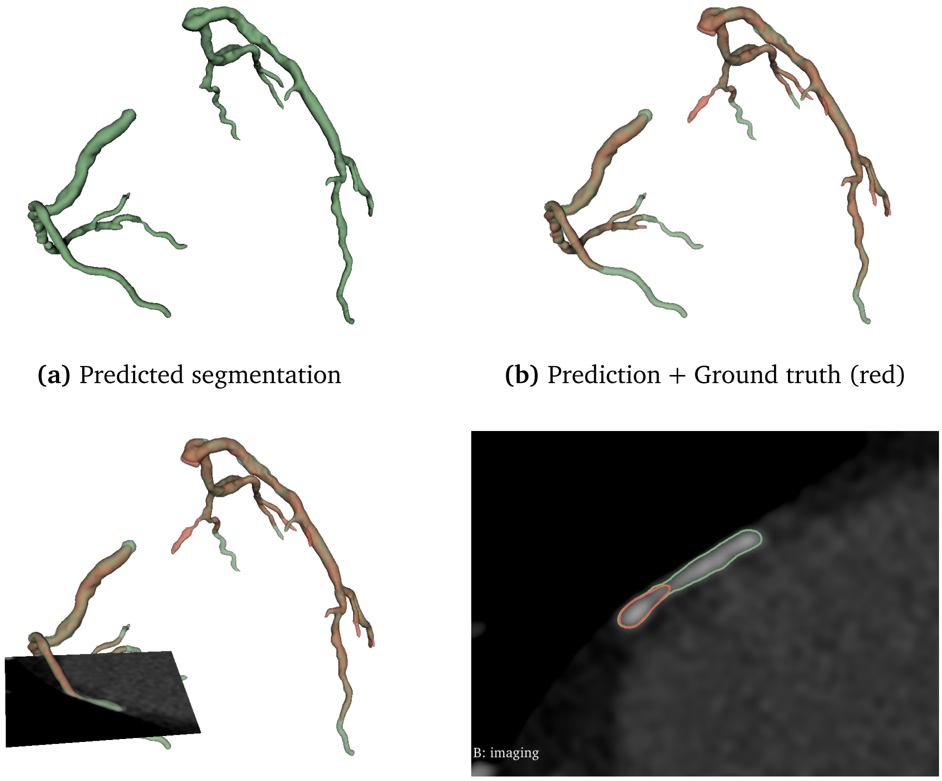
(c) Prediction (green), Ground truth (red) and stenosis, grade 3 (violet)

(d) CCTA (Axial plane) overlaid on 3D view

(e) CCTA (Axial plane) with segmentation contours

Figure 4.5: The disjointed segmentation (green) is compared with ground truth (red) and stenosis (violet) in both 3D view, and in the axial plane of the raw CCTA data.

The last example shows a sample where the post-processed prediction has false positives in some distal parts of the arteries when compared to the ground truth, see Figure 4.6b. The CCTA raw data shows that the "false positives" are related to high contrast areas, see Figure 4.6d. See link for a video demonstration of the region of interest.



(a) Predicted segmentation

(b) Prediction + Ground truth (red)

(c) Prediction + Ground truth with CCTA overlay (Axial plane)

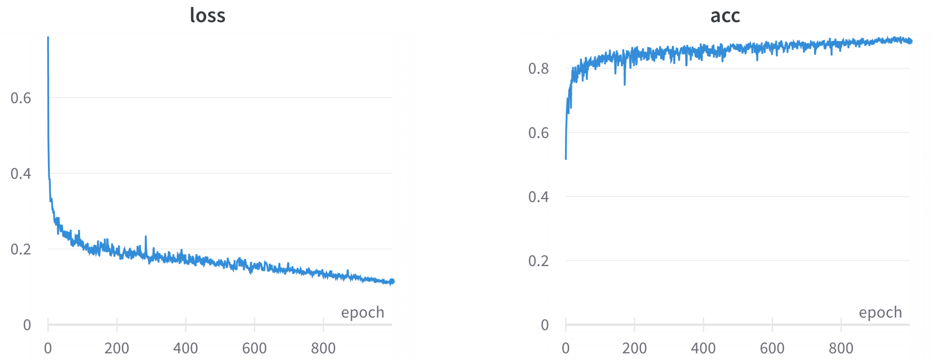
(d) CCTA (Axial plane) with segmentation contours

Figure 4.6: The segmentation (green) is compared with ground truth (red) in both 3D view and in the coronal plane of the raw CCTA data.

4.4 Experiment 4: nnU-Net vs SWIN UNETR

In this experiment a nnU-Net model was trained on the St. Olavs Hospital dataset and then compared with the results from the best SWIN UNETR model from experiment 1. The reason for not using the fine-tuned SWIN UNETR model was due to the increased number of training samples, leading to an unfair comparison. Figure 4.7 shows the training loss and validation accuracy during training of the nnU-Net model. The best validation DSC during training was 0.8999 for this model, note that this score is calculated per patch with a size of 160 x 96 x 160 unlike the SWIN UNETR validation, which was calculated on the full volume.

In order to properly evaluate the nnU-Net model against the SWIN UNETR model,

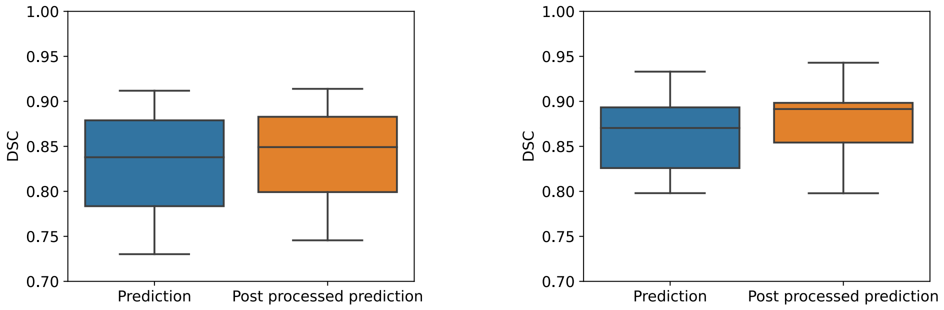


(a) Training loss

(b) Validation accuracy (DSC)

Figure 4.7: Training loss and validation accuracy for nnU-Net model
Note that the validation accuracy is only per patch of size 160 x 96 x 160.

both the pure prediction and the post-processed prediction were calculated using the same MONAI evaluation code. See Figure 4.8 for the distribution of DSC for both models for all samples in the St. Olavs Hospital dataset test set. Table 4.5 shows the accuracy for the predictions as well as post-processed predictions for each of the 12 samples in the test set for each of the two models.



(a) nnU-Net

(b) SWIN UNETR

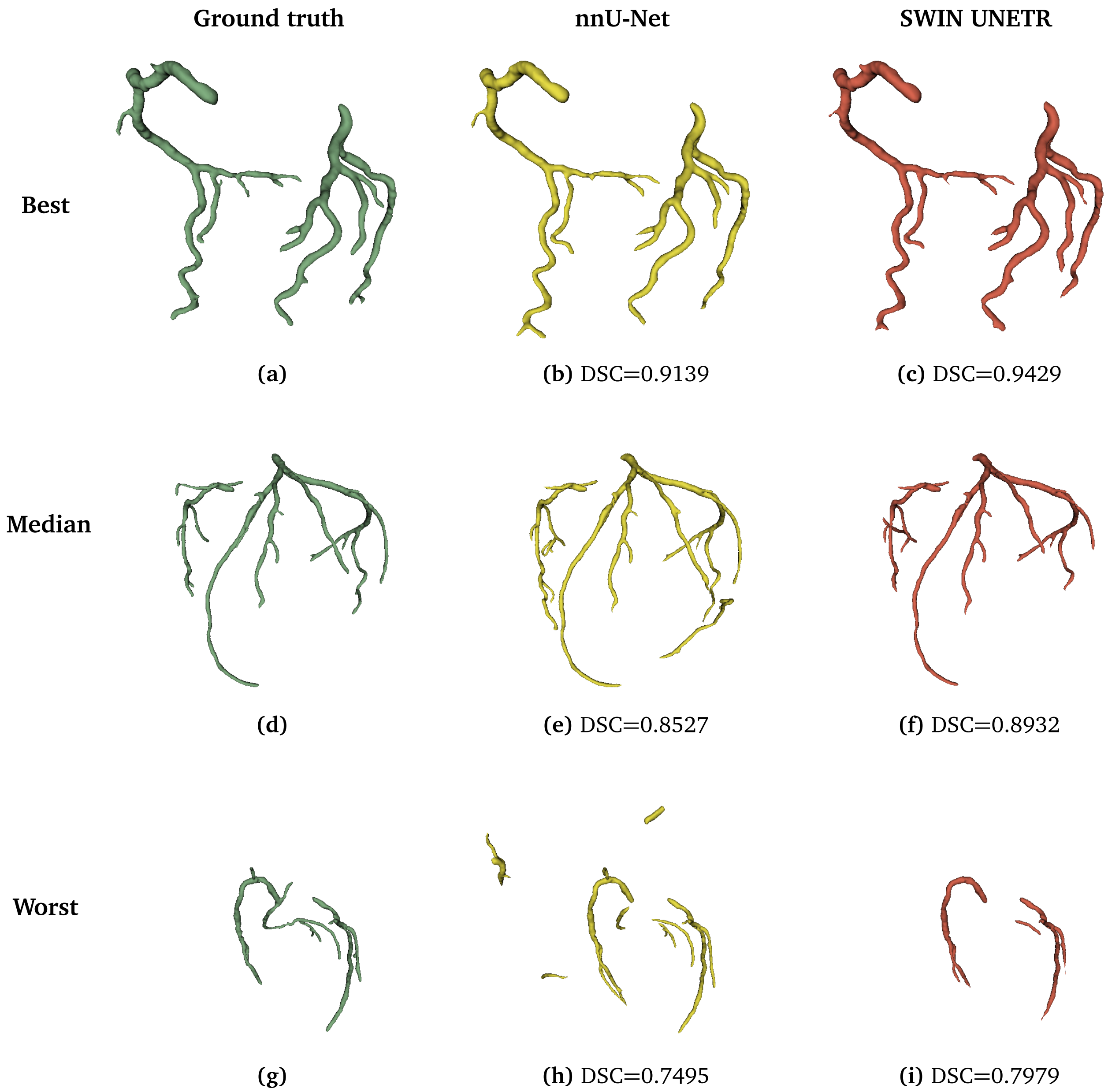
Figure 4.8: nnU-Net vs SWIN UNETR accuracy

	nnU-Net				**SWIN UNETR**			
Sample	DSC	DSC post	HD 95	HD 95 post	DSC	DSC post	HD 95	HD 95 post
1	0.7653	0.7883	33.42	32.58	0.8189	0.8336	20.09	16.05
2	0.8315	0.8456	24.46	21.86	0.7990	0.8220	57.30	51.99
3	0.7896	0.8028	73.00	73.41	0.8585	0.8897	41.12	2.40
4	0.7421	0.7456	36.39	36.06	0.8282	0.8621	20.09	8.93
5	0.9118	0.9139	0.76	0.47	0.9330	0.9429	2.03	0.47
6	0.8739	0.8775	2.18	1.09	0.8533	0.8611	36.12	4.42
7	0.7302	0.7495	76.84	77.31	0.7980	0.7979	56.99	18.77
8	0.8622	0.8733	38.14	27.04	0.8952	0.8981	3.32	2.04
9	0.894	0.9075	7.34	1.59	0.9120	0.9249	7.55	0.89
10	0.8139	0.8168	37.59	38.13	0.8823	0.8990	11.65	3.82
11	0.8443	0.8527	25.33	22.80	0.8888	0.8932	8.86	7.61
12	0.8993	0.8989	7.31	6.60	0.8927	0.8943	7.83	6.06
Mean	0.8298	0.8394	30.23	28.24	0.8633	**0.8766**	22.75	**10.29**
STD	0.0595	0.0562	24.03	24.81	0.0427	**0.0408**	19.31	**13.73**

Table 4.5: DSC and 95th percentile Hausdorff distance (mm) for nnU-Net and SWIN UNETR model predictions and post processed predictions on the St. Olavs Hospital dataset.

4.4.1 Qualitative Comparison

In order to fully evaluate the differences in the predictions between nnU-Net and SWIN UNETR a selection of predictions from the test set was analyzed. Figure 4.9 shows the best, median and worst prediction based on DSC with respect to the post processed predictions from SWIN UNETR. In Figure 4.9h it is apparent that the post-processing threshold of a minimum of 1000 voxels was not sufficient to remove all artifacts.



(a) **(b)** DSC=0.9139 **(c)** DSC=0.9429

(d) **(e)** DSC=0.8527 **(f)** DSC=0.8932

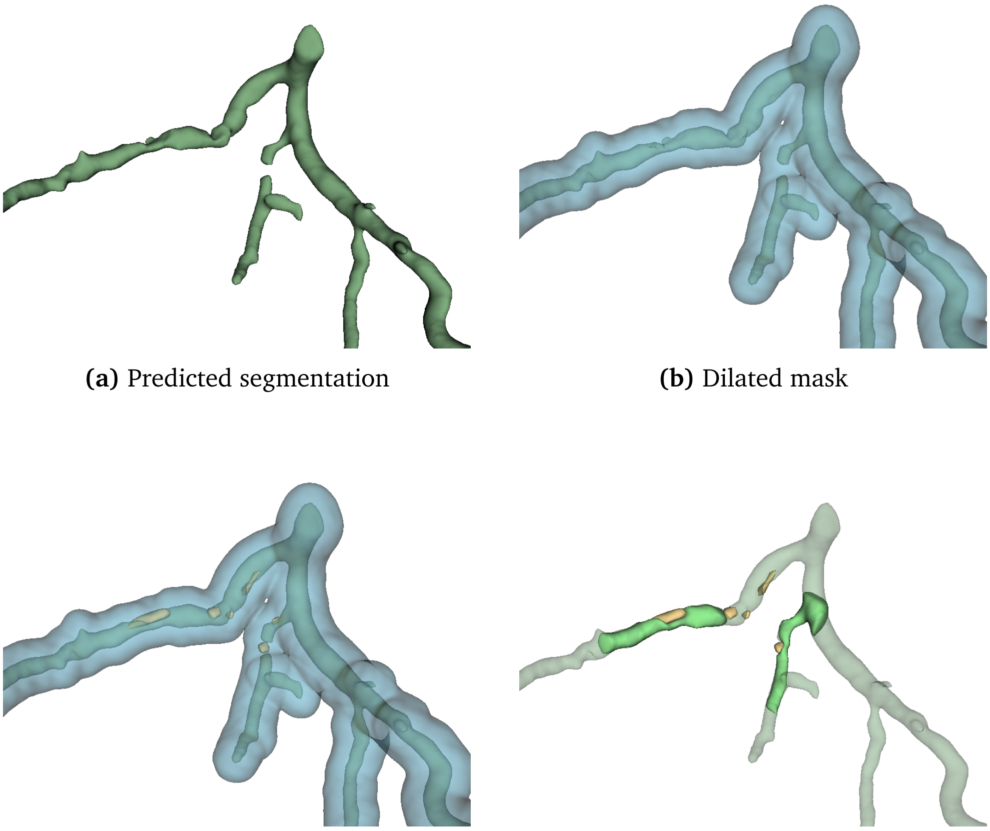
(g) **(h)** DSC=0.7495 **(i)** DSC=0.7979

Figure 4.9: Qualitative comparison between SWIN UNETR and nnU-Net predictions. The rows show the best, median and worst prediction based on SWIN UNETR DSC on the test set. The columns shows the ground truth, nnU-Net and SWIN UNETR post processed prediction

4.5 Experiment 5: Plaque localization

In this experiment, the plaque localization post-analysis was performed. Although there are radiodensity overlaps between the luminal contrast and the different types of plaque, calcified plaque does span above the normal range of luminal contrast range. Using the information from the dataset statistics Table 3.2 a threshold of 850HU was selected for localizing calcified plaque without interfering with the typical luminal contrasts. In addition to the HU threshold, an assumption was made that the plaque should be in close proximity to the segmented coronary arteries.

This method is executed by dilating the prediction and selecting all voxels above the threshold that resides inside the dilated mask. Figure 4.10 shows a visual representation of the pipeline, where green is the predicted segmentation, blue is the dilated mask and plaque is yellow. Finally, the plaque location is compared with the clinically localized stenosis location in light green.



(a) Predicted segmentation

(b) Dilated mask

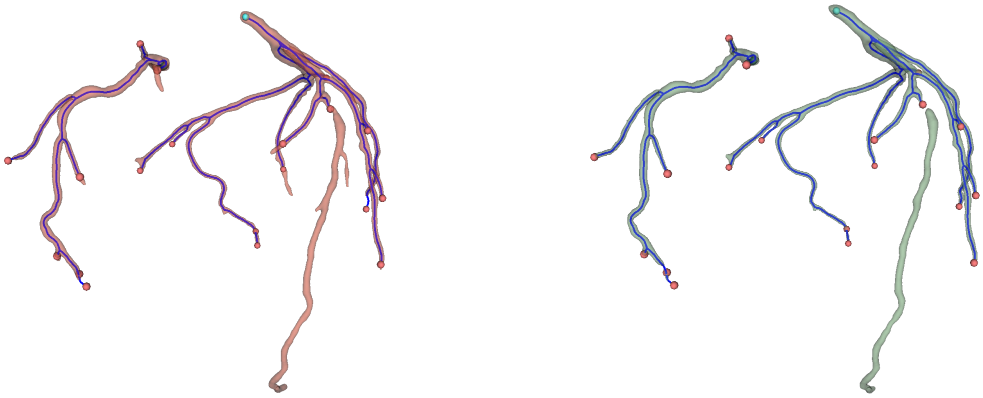
(c) Plaque

(d) Clinically localized stenosis

Figure 4.10: Visual representation of the plaque localization pipeline.(b) shows how the dilated mask is related to the post-processed prediction and represents the area that is used to perform the plaque localization. (c) shows the resulting plaque within the dilated mask (HU > 850). and (d) showing the ground truth stenosis in relation to the plaque. Notice how the post-processed prediction, fails to connect the segmentation in the same place as the stenosis when comparing (a) and (d).

4.6 Experiment 6: Centerline Extraction

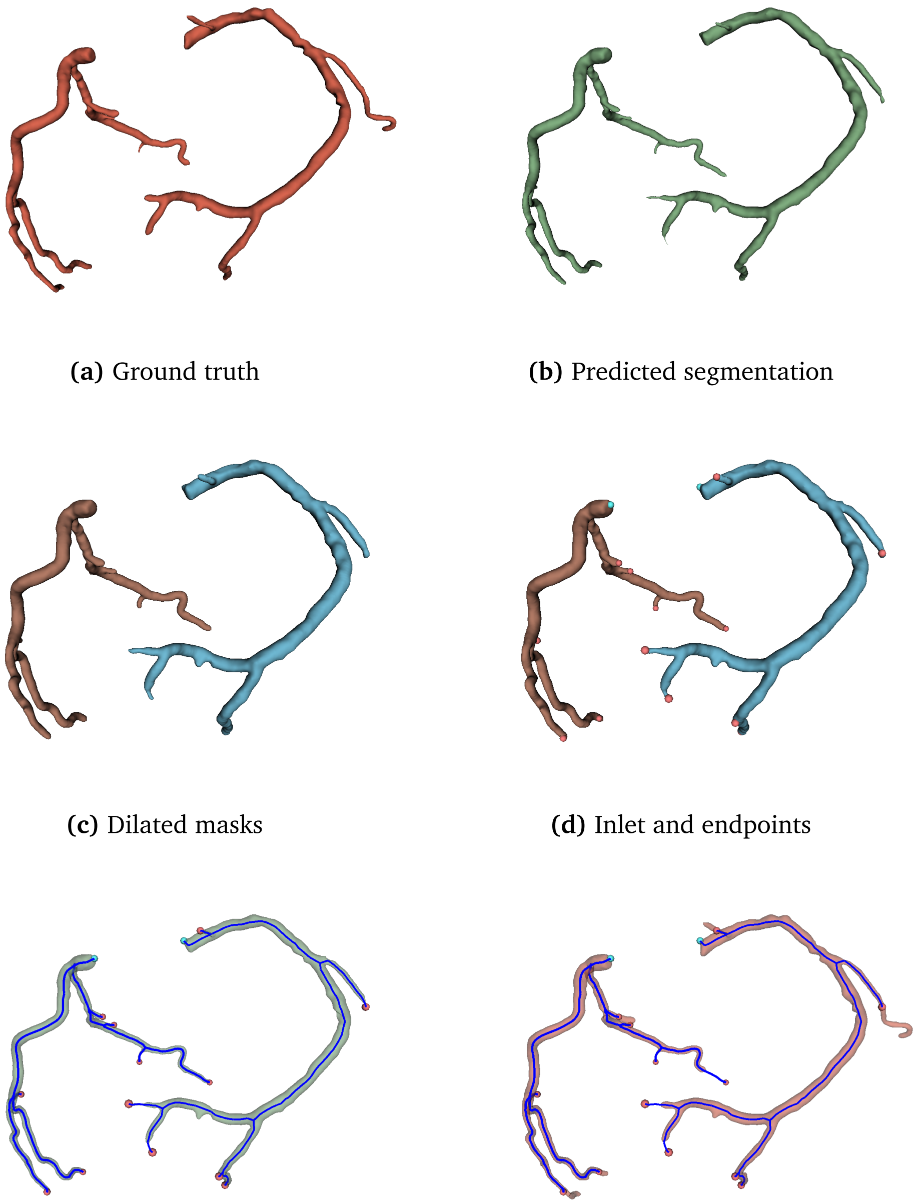
In this experiment, the centerline extraction pipeline based on VMTK and 3D Slicer was performed. The centerline extraction was performed on post-processed predicted images from the St.Olavs Hospital dataset, with the corresponding fine-tuned SWIN UNETR Model. See Section 3.4.2 for more information about the methodology. Figure 4.12 shows a visual representation of the full centerline extraction, on a post-processed prediction from the fine-tuned SWIN UNETR model. In this sample, the pipeline successfully extracted the centerline, but due to some false negatives distally, the centerline is not fully complete, see Figure 4.12f. In this example, it is also worth noting that the issue with false positives leading to disjointed segmentation did not occur. In the predictions where segmentation was sufficiently disjointed to the point that the dilated mask could not reconnect the areas, the centerline did get cut off at the disjointed location, which in turn led to an incomplete centerline, see Figure 4.11.



(a) Ground truth

(b) Post processed prediction

Figure 4.11: Incomplete centerline extraction



(a) Ground truth

(b) Predicted segmentation

(c) Dilated masks

(d) Inlet and endpoints

(e) Centerline over prediction

(f) Centerline over GT

Figure 4.12:

4.7 Experiment 7: FFR prediction

In this experiment, FFR predictions were performed by utilizing a trained ANN with ROM input features from previous work [17]. The St. Olavs Hospital dataset was used for this experiment as it also contains clinically assessed stenoses, as well as physically measured FFR for comparison. In the St. Olavs Hospital test set, two samples were excluded due to one sample having issues with the FFR measurement and the other sample was excluded due to an abnormal coronary artery tree. The predicted segmentations used as input for the FFR estimation network in this experiment were generated by using the SWIN UNETR model that was fine-tuned on the St. Olavs Hospital dataset. Table 4.7 shows the results for each of the three variations for the FFR estimation using ground truth, post-processed prediction and the manually corrected prediction as input to the ANN.

The statistical analysis of the FFR estimation results shows that using ground truth as the input gave the highest correlation to the measured FFR. The average error was lowest when using the manually corrected prediction as the input, but the same input also gave the highest standard deviation error of the three alternatives. The key threshold for assessing functionally significant stenosis is set at $FFR < 0.8$. Using this threshold as a binary classification, the FFR estimation using the ground truth and the manually corrected prediction, managed to correctly classify functionally significant stenosis with an 83.3% accuracy. The modified predictions did however outperform the ground truth in sensitivity with a 85.7% vs 71.4%. See Table 4.6 for additional statistical findings.

	FFR_{GT}	FFR_{pred}	$FFR_{modified}$
Bias	-0.044	-0.069	**-0.037**
STD	**0.1**	0.124	0.126
a	0.544	0.43	0.442
b	0.399	0.514	0.472
r	**0.87**	0.756	0.731
r2	**0.639**	0.39	0.478
Accuracy	**83.3**	72.2	**83.3**
Sensitivity	71.4	57.1	**85.7**
Specificity	**90.9**	81.8	81.8
AUC	**0.883**	0.714	0.851

Table 4.6: FFR estimation statistics

Sample	Branch	FFR	FFR_{GT}	FFR_{pred}	$FFR_{modified}$
1	LM	0.45	0.58	0.61	0.60
1	LM	0.34	0.58	0.67	0.65
1	RCA	0.53	0.73	0.74	0.74
3	LM	0.77	0.76	0.80	0.75
3	LM	0.87	0.90	0.77	0.77
4	LM	0.85	0.82	0.91	0.91
4	LM	0.74	0.88	0.99	0.92
4	RCA	0.92	0.91	0.93	0.93
5	LM	0.96	0.93	0.90	0.91
6	LM	0.83	0.71	0.77	0.83
6	LM	0.94	0.84	0.98	0.67
8	LM	0.64	0.75	0.78	0.77
8	LM	0.95	0.97	0.94	0.96
9	LM	0.9	0.89	0.90	0.88
10	LM	0.82	0.88	0.85	0.86
10	LM	0.955	0.98	0.94	0.97
11	LM	0.64	0.81	0.91	0.70
12	LM	0.92	0.89	0.88	0.86

Table 4.7: FFR estimation results

FFR is the actual measured value for each of the stenoses, where some samples contain more than one measurement. Branch refers to which branch of the coronary artery the measurement is associated with. The FFR subscripts refer to what input was used for estimating FFR, namely ground truth (GT), post-processed prediction (pred) and manually corrected predictions (modified).

Chapter 5

Discussion

In this chapter, the results from the experiments conducted in this thesis, as well as the potential shortcomings of the methodology are discussed in relation to the research questions.

5.1 Experiments

Experiment 1: Hyper-parameter Selection In the first experiment, a selection of hyper-parameters was used to train the SWIN UNETR network in order to select the best configuration before further training. The selection of hyper-parameters in this experiment was selected on the basis of the extreme class imbalance present in the dataset, as well as the inclusion of SSL pre-trained weights from another medical domain.

Although both Dice loss and focal loss address the issue of class imbalance, focal loss achieved a slightly increased DSC of 0.8127 vs 0.7940. The reason focal loss achieved a better result could be due to the way it focuses on the voxels that the model is unsure of, which could contribute to more focus on the border between the arteries and the background as well as harder to predict voxels within the coronary arteries. The second measure for dealing with class imbalance was to oversample patches with either a foreground voxel or a background voxel as the center with a set probability, where the baseline probability was set at 50%. Increasing the probability to 75% for the center being a foreground voxel positively affected the DSC further 0.8329 vs 0.8127.

The last hyper-parameter checked was if using pre-trained weights using SSL on 5050 samples of medical CT data would increase the model accuracy. Although none of the datasets contained specifically coronary artery segmentations, some of the CT scans would presumably contain the coronary arteries as some of the datasets are focused on lung segmentation. The SWIN UNETR network was trained with the pre-trained weights with and without frozen encoder weights where the results showed an increased DSC of 0.8447 with unfrozen encoder and decreased with frozen encoder (0.6824).

Although this experiment only tested a few hyper-parameters out of many possibilities, the selection of the hyper-parameters did have a significant positive impact on the model DSC. The feature size of the SWIN UNETR network is another area that could be tested, but as the SSL pre-trained weights from earlier work were trained with a feature size of 48 we kept the feature size at the same in order to utilize these weights.

Experiment 2: Baseline SWIN UNETR model

In the field of medical image analysis, the lack of a sufficient amount of training data is often an issue when training DL networks. The lack of data is due to the time-consuming nature of labeling as well as privacy laws blocking the distribution of the data. In order to ensure that the SWIN UNETR model was trained on a sufficient amount of data, all the three datasets discussed in this thesis were combined in order to train a baseline SWIN UNETR model with as many samples as possible. As the largest dataset contains 85% of the samples in the combined dataset, this combination should be very impactful when contrasted to training the model for each dataset separately.

Using the best hyper-parameters discovered from the first experiment a final DSC of 0.8265 was achieved when training on the combined dataset. This DSC was lower than the accuracy on the hyper-parameter selection experiment but could be due to the increased amount of data, and the fact that the three datasets might differ slightly in the segmentation strategy as well as the imaging quality. Another thing to note is that this model could perhaps benefit from further training, as the DSC did not show a complete sign of convergence (Figure 4.1).

Experiment 3: Fine-tuned models

As each of the datasets could differ slightly, an additional step of training a separate fine-tuned model for each of the three datasets was performed in order to maximize the DSC using the baseline model as the starting point. Arguably this step would actually make the model less generalized, but serves as a good tool for maximizing the accuracy before comparing it to previous work. By comparing the results from the hyper-parameter search experiment to the fine-tuned model trained with the St. Olavs Hospital dataset, it is apparent that using the baseline SWIN UNETR model as the starting point for fine-tuning did indeed perform better, with a DSC 0.8657 vs 0.8447.

Despite the ImageCAS dataset making up 85% of the total combined dataset, a substantial enhancement was noted during the fine-tuning phase on this specific dataset. The improvement was visible when comparing the DSC scores of the baseline model at 0.8265 and the fine-tuned model, which achieved a significantly higher score of 0.8528. One possible explanation for this big improvement in accuracy could be the narrow window of selecting the 0.05 and 99.5 percentile of the foreground intensity for a clipped normalization while training the baseline model. This normalization interval was increased to the maximum and minimum of the label intensity based on the St. Olavs Hospital due to an increased model accuracy, which might explain the big gap in model performance on the ImageCas dataset between the baseline and the fine-tuned model.

For evaluating the three models on the corresponding test sets, post-processing and exclusion of the voxel spacing normalization did yet again increase the DSC for all the three models, which explains why the test set DSC are higher than the validation DSC. Given these final test set predictions DSC, the model fine-tuned on ImageCAS outperformed the baseline patch based CNN method presented by the ImageCAS paper [9] by a large margin, 0.8614 vs 0.8296. It is however worth noting that the test set used in the paper consisted of 25% of the total samples compared to the 10% used in this thesis, This difference might account for some of the difference but due to the size of the dataset (1024 samples), this difference should not affect the score too much.

The test set for the fine-tuned model trained on the Coronary Atlas was evaluated on the ASOCA Challenge website. In this submission, the final DSC ended up at 0.8663 and ended up in the 7th place 0.0193 behind the top submission at 0.8856. Compared to the paper published about the challenge our score ties with the top submission in terms of DSC (0.87), but had a marginally higher standard deviation of 0.05 vs 0.04, this comparison is however limited to two decimal places as that is the rounding used in the original publication. One thing to note about this dataset is the number of samples was significantly lower than the ImageCAS dataset (60 vs 1024 samples).

In the visual analysis of the predicted segmentations it was revealed that the model predicted FP distally of the coronary arteries in multiple samples when compared to ground truth. Inspecting these areas further in the raw CCTA, revealed that most of these areas had a clear luminal contrast, indicating the presence of coronary arteries. This difference between the prediction and ground truth suggests that the radiologists intentionally dropped these parts of the artery tree due to not being clinically relevant, dropped them due to their cross-section area or simply missed them during manual segmentation. This inconsistency between the ground truth and prediction contributes to a lower DSC, but actually indicates a better segmentation than ground truth, and does not contribute to any problems with the FFR estimation.

Some problematic areas in the predicted segmentations did also get revealed by the visual analysis, where instances of false negative predictions lead to disjointed segmentations. By analysis of the raw data, these locations often correlated to either stenosis or a reduced luminal contrast, which makes it understandable that the model might struggle in these areas.

Experiment 4: nnU-Net vs SWIN UNETR

In the fourth experiment, nnU-Net was trained on the St. Olavs Hospital dataset, with results compared to the SWIN UNETR model. The best-performing SWIN UNETR model from the first experiment was selected to ensure equal training data for both models. Although solely trained on the St. Olavs Hospital dataset, SWIN UNETR incorporated SSL pre-trained weights.

Upon evaluating the post-processed test set predictions, nnU-Net demonstrated lower performance than the SWIN UNETR model, with a DSC of 0.8394 versus 0.8765. Analyzing the 95th percentile for the Hausdorff distance, a considerably larger boundary discrepancy was observed between nnU-Net's predictions and the ground truth, as compared to SWIN UNETR (28.24mm vs 10.29mm). Notably, even after the post-processing steps, which removed segments smaller than 1000 voxels from both models' predictions, the nnU-Net produced more artifacts above this threshold. In contrast, the SWIN UNETR predictions contained fewer artifacts. This variance in the number of large artifacts might explain some of the differences in the Hausdorff distance, as the model predictions were quite similar visually apart from this aspect.

Experiment 5: Plaque Localization

By utilizing the domain knowledge about typical HU intensities for plaque and the inner parts of the coronary arteries (lumen), a pipeline was designed in order to localize plaque. By setting the intensity threshold to 850HU and by performing a dilation on the predicted segmentation, the results showed that the resulting plaque segmentation was successful. By analyzing the visual results, it was apparent that the plaque had almost no overlap with the predicted coronary artery segmentation, which signalizes the prediction does a good job of not falsely classifying plaque as lumen. In addition, the segmented plaque was also present in most areas around the clinically segmented stenosis areas. This correlation between the plaque and the stenosis is a good sign that the pipeline is working, as plaque is the cause of stenosis.

As the intensities for plaque and coronary lumen do have a bit of overlap, the type of plaque segmented in with this pipeline was limited to high-intensity calcified plaque (850HU +). The observed correlation between the disjointed predicted segmentations and the presence of calcified plaque does however indicate that the SWIN UNETR model is struggling with these ambiguous intensity ranges.

Experiment 6: Centerline Extraction

As mentioned in Section 2.8.4, in the previous work on FFR estimation, the centerline model needed to perform the estimation was not automatically generated. In order to achieve a fully automatic FFR estimation, an automatic centerline extraction was designed and performed in this experiment.

The automatic centerline extraction was designed using the 3D Slicer Python environment in order to perform automatic localization of the inlets and outlets associated with each of the two coronary arteries. Using these points, the centerline model was successfully produced by using VMTK.

By analyzing the results visually, it was apparent that the issues presented in experiment 4, namely the disjointed segmentations, caused an incomplete centerline extraction. This issue was to be expected as the centerline only uses the two largest connected segments as the input.

Experiment 7: FFR Prediction

In the last experiment, segmentations produced by the SWIN UNETR model, trained on the St. Olavs Hospital dataset was used as input to the FFR estimation network from earlier work. The St. Olavs Hospital dataset was used as it was the only dataset out of the three, that had clinically segmented stenosis and associated physical FFR measurements included. Having the physically measured FFR and the location allowed for a proper evaluation of the accuracy of the estimated FFR predictions.

In addition to the predicted segmentations, the FFR network also needs the centerline model, which was produced by the automatic centerline extraction method proposed in this thesis. Unfortunately, the disjointed locations in the predicted segmentations pose a problem for the centerline extraction which leads to an incomplete centerline model as seen in Figure 4.11. As the FFR estimation network uses the geometry along the centerline, the FFR estimation also becomes incomplete. If the disjointed section is not cutting off the region of interest, this is not a problem for the FFR prediction. Unfortunately, the disjointed segmentation did cut off important areas in multiple cases.

In addition, to post-processed predictions, and ground truth segmentation, a third set of manually corrected segmentations was also used as input to the centerline extraction method and the FFR prediction network. Although correcting the predictions requires manual work, the amount of work needed is dramatically reduced if compared to segmenting the arteries from scratch. The results from the FFR estimation unsurprisingly showed that the best correlation to the physical FFR measurements were achieved by using ground truth as input.

The most interesting question in measuring FFR is if the area is associated with functionally significant stenosis which is defined as FFR < 0.8. Using this classification, the manually corrected segmentation outperforms the classification made with ground truth as input, with 85.7% vs 71.4% in its ability to correctly classify a given location as functionally significant stenosis (sensitivity). In contrast the uncorrected prediction with a sensitivity of 57.1%, which is obviously insufficient as misclassification of functionally significant stenosis instances could be fatal. Note that if the predicted segmentation was disjointed proximal to the stenosis location, the measurement was compared to the closest estimated FFR value proximal to the measured value.

5.2 Research Questions

5.2.1 RQ 1: How does recent transformer-based architectures compare with current CNN-based methods for segmentation of coronary arteries?

SWIN UNETR was identified through literature review as the current SOTA for various medical segmentation tasks. However, no public information was found on SWIN UNETR's accuracy on the task of coronary artery segmentation. This lack of published data inspired the selection of experiments performed in this thesis. With the goal of comparing the SWIN UNETR model accuracy for segmentation of coronary arteries with previously published results, experiments were conducted in order to find the best configuration for training by addressing class imbalance and by maximizing the use of available training data. Finally, a baseline SWIN UNETR model was fine-tuned on each of the three datasets and compared the results with previous work. In this comparison, the fine-tuned SWIN UNETR model exceeded the accuracy presented in the ImageCAS paper, with a DSC of 0.8614 vs 0.8296 [9]. In the ASOCA Challange [8] the fine-tuned SWIN UNETR managed to get 7th place in the online leaderboard with a DSC of 0.8663, marginally lower than the top submission with a DSC of 0.8856. Additionally, in relation to the published results on the ASOCA Challenge, our submission tied with the top submission with a DSC of 0.87 [8], given the rounding to two decimal places used in the published results.

As the St. Olavs Hospital dataset did not have any previously published segmentation scores to compare to, nnU-Net was trained and compared to the non-fine tuned SWIN UNETR model. In this comparison, the mean DSC of the SWIN UNETR model did outperform the results from nnU-Net. It was also observed that the nnU-Net model predictions had considerably more large artifacts in the model predictions, compared to SWIN UNETR. The presence of these large artifacts was also reflected in the mean 95th percentile Hausdorff distance, where the boundary discrepancy was more than double in nnU-Net than SWIN UNETR for the post-processed predictions.

5.2.2 RQ 2: Is it possible to combine automatic coronary artery segmentation with previous work on FFR estimation for clinical assessment of CAD?

Combining automatic coronary artery segmentation with previous FFR estimation work promises exciting opportunities in the realm of clinical CAD assessment. The SWIN UNETR model, in particular, delivered compelling results. While a few challenges were encountered, such as disjointed areas in some samples, the overarching quality of segmentation was notably high. In fact, the SWIN UNETR model's performance outperformed that of the manually segmented samples, which served as the ground truth, in several instances.

Even though some problem areas required manual correction, they were easily identifiable, and the corrections were straightforward to implement. Consequently, a set of these manually corrected predictions was fed into the FFR prediction network. Resulting in an impressive correlation between our predicted FFR values and those physically measured.

Applying the estimated FFR to classify stenosis as functionally significant (FFR < 0.8), promising sensitivity of 85.7% compared to physical measurements was attained. This sensitivity exceeded the performance achieved when using clinically segmented arteries as input. However, the sensitivity, dipped to 57.1% when using non-corrected segmentations.

The automatic centerline extraction, presented in this paper, facilitates a fully automated process for FFR estimation from CCTA images. While the present SWIN UNETR model does necessitate manual corrections to reach optimal diagnostic accuracy, these corrections are significantly less labor-intensive than a full manual segmentation. In light of this considerable reduction in workload, this method could serve as an invaluable asset to clinicians assessing CAD, enhancing both their efficiency and diagnostic precision.

5.3 Reflection

This section provides a reflective appraisal of the thesis journey, identifying some negative aspects that would have been performed differently if done again.

One negative aspect that could have been improved upon in writing this thesis, was the insufficient time used for understanding the subject of CAD, FFR and medical imaging at the beginning of the project. Using more time for understanding the domain would have reduced the number of pitfalls, and confusion while implementing numerous parts of the segmentation pipelines, as well as the writing.

Although preliminary research was performed in order to gather additional relevant datasets, this endeavor was unsuccessful. The ImageCAS and Coronary Atlas were acquired eventually, but more persistence could have been used in order to acquire these at an earlier stage.

Another aspect that hindered the development of the experiments and understanding was the reliance on previously written code for SWIN UNETR. In the middle of this project, the SWIN UNETR code was completely rewritten in PyTorch Lightning, which greatly simplified the code and allowed for a better understanding when conducting experiments.

Chapter 6

Conclusion and Further Work

In this thesis, the State of The Art (SOTA) transformer model Shifted Window U-Net Transformer (SWIN UNETR) was trained on the task of coronary artery segmentation, in order to assess the performance compared to current Convolutional Neural Network (CNN)-methods. Furthermore, the potential for combining automatic artery segmentation with Fractional Flow Reserve (FFR) estimation in a clinical context was examined. In this chapter, the two research questions are concluded.

RQ 1: How does recent transformer-based architectures compare with current CNN-based methods for segmentation of coronary arteries? The transformer-based SWIN UNETR architecture demonstrated superior performance over CNN-based methods such as no new U-Net (nnU-Net), indicating that transformer-based architectures could potentially set a new standard in coronary artery segmentation.

RQ 2: Is it possible to combine automatic coronary artery segmentation with previous work on FFR estimation for clinical assessment of CAD? The results suggested that combining automatic artery segmentation with FFR estimation for Coronary Artery Disease (CAD) assessment is feasible. Using SWIN UNETR segmentation with FFR estimation showed a high correlation between predicted and physically measured FFR. Additionally, the diagnostic sensitivity of identifying functionally significant stenosis was enhanced, compared to using ground truth segmentations as input. Manual corrections at disjointed areas of the predicted segmentations were however needed in order to get these results. However, manual corrections were only needed in certain cases and in specific regions so the overall workload was significantly reduced, indicating the possibility of being a valuable tool for clinicians in diagnosing CAD.

Further work is needed to improve segmentation quality, particularly in disjointed areas, and to implement a streamlined FFR estimation pipeline for clinical use.

6.1 Further Work

In order to perform the FFR estimation directly from the SWIN UNETR predictions, additional work needs to be done in order to fix the issue of disjointed segmentations. As the experimental results in this thesis showed a positive impact of oversampling positive labels, a similar tactic could be utilized to oversample samples directly from these problematic areas for training. Another possibility could be to pre-process the training data with filters like the Frangi vesselness filter in order to improve the luminal contrast.

As mentioned in the previous section, the full FFR estimation from Coronary Computed Tomography Angiography (CCTA) images could be fully automated as none of the components needs manual interaction, in order to function. In order to actually implement this full pipeline, more work is needed in order to fit these pieces together. Given that it is possible to fix the issue of disjointed segmentations, this pipeline could save clinicians a lot of work. If it is not possible to sufficiently train the model to remove the issue of disjointed segmentations, the parts can still be combined as a valuable clinical tool. One possible design solution could be a pipeline using 3D Slicer in conjunction with Monai label in order to accommodate inference, manual corrections, centerline extraction and finally the FFR estimation all in one unified GUI. As 3D Slicer is open-source and provides a Python API, this integration should not be too difficult. The resulting product should potentially be much easier to use than using the individual parts by themselves.

If this full pipeline, either by using manual corrections or fully automatic were to be used in a clinical environment, additional clinical trials need to be performed in order to assess the clinical usefulness.

Bibliography

[1] G. A. Roth, D. Abate, K. H. Abate, S. M. Abay, C. Abbafati, N. Abbasi, H. Abbastabar, F. Abd-Allah, J. Abdela, A. Abdelalim and et al., 'Global, regional, and national age-sex-specific mortality for 282 causes of death in 195 countries and territories, 1980–2017: A systematic analysis for the global burden of disease study 2017,' *The Lancet*, vol. 392, no. 10159, pp. 1736–1788, 2018. DOI: `10.1016/s0140-6736(18)32203-7`.

[2] G. Montalescot, U. Sechtem, S. Achenbach, F. Andreotti, C. Arden, A. Budaj, R. Bugiardini, F. Crea, T. Cuisset, C. Di Mario *et al.*, '2013 esc guidelines on the management of stable coronary artery disease: The task force on the management of stable coronary artery disease of the european society of cardiology,' *European heart journal*, vol. 34, no. 38, pp. 2949–3003, 2013.

[3] M. Tavakol, S. Ashraf and S. J. Brener, 'Risks and complications of coronary angiography: A comprehensive review,' *Global Journal of Health Science*, vol. 4, no. 1, 2011. DOI: `10.5539/gjhs.v4n1p65`.

[4] *Mimics*. [Online]. Available: `https://www.materialise.com/en/healthcare/mimics-innovation-suite/mimics` (visited on 02/06/2023).

[5] *3d slicer*. [Online]. Available: `https://www.slicer.org/` (visited on 06/02/2023).

[6] *Itk-snap*. [Online]. Available: `http://www.itksnap.org/` (visited on 14/06/2023).

[7] P. A. Yushkevich, J. Piven, H. Cody Hazlett, R. Gimpel Smith, S. Ho, J. C. Gee and G. Gerig, 'User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability,' *Neuroimage*, vol. 31, no. 3, pp. 1116–1128, 2006.

[8] G. Tetteh, T. Preußer, V. Efremov, N. D. Forkert, A. Schlaefer, A. Lundervold and G. Wu, 'Asoca: Automatic segmentation of coronary arteries,' in *International Workshop on Statistical Atlases and Computational Models of the Heart*, Springer, 2020, pp. 130–140.

[9] A. Zeng, C. Wu, M. Huang, J. Zhuang, S. Bi, D. Pan, N. Ullah, K. N. Khan, T. Wang, Y. Shi, X. Li, G. Lin and X. Xu, *Imagecas: A large-scale dataset and benchmark for coronary artery segmentation based on computed tomography angiography images*, 2022. arXiv: `2211.01607 [eess.IV]`.

[10] O. Ronneberger, P. Fischer and T. Brox, *U-net: Convolutional networks for biomedical image segmentation*, 2015. DOI: 10.48550/ARXIV.1505.04597. [Online]. Available: https://arxiv.org/abs/1505.04597.

[11] F. Isensee, P. F. Jaeger, S. A. Kohl, J. Petersen and K. H. Maier-Hein, 'Nnu-net: A self-configuring method for deep learning-based biomedical image segmentation,' *Nature Methods*, vol. 18, no. 2, pp. 203–211, 2020. DOI: 10.1038/s41592-020-01008-z.

[12] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh and J. Liang, 'Unet++: A nested u-net architecture for medical image segmentation,' in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, D. Stoyanov, Z. Taylor, G. Carneiro, T. Syeda-Mahmood, A. Martel, L. Maier-Hein, J. M. R. Tavares, A. Bradley, J. P. Papa, V. Belagiannis, J. C. Nascimento, Z. Lu, S. Conjeti, M. Moradi, H. Greenspan and A. Madabhushi, Eds., Cham: Springer International Publishing, 2018, pp. 3–11, ISBN: 978-3-030-00889-5.

[13] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox and O. Ronneberger, '3d u-net: Learning dense volumetric segmentation from sparse annotation,' *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 424–432, 2016.

[14] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit and N. Houlsby, *An image is worth 16x16 words: Transformers for image recognition at scale*, 2020. DOI: 10.48550/ARXIV.2010.11929. [Online]. Available: https://arxiv.org/abs/2010.11929.

[15] A. Hatamizadeh, V. Nath, Y. Tang, D. Yang, H. Roth and D. Xu, *Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images*, 2022. DOI: 10.48550/ARXIV.2201.01266. [Online]. Available: https://arxiv.org/abs/2201.01266.

[16] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin and B. Guo, *Swin transformer: Hierarchical vision transformer using shifted windows*, 2021. DOI: 10.48550/ARXIV.2103.14030. [Online]. Available: https://arxiv.org/abs/2103.14030.

[17] F. E. Fossan, L. O. Müller, J. Sturdy, A. T. Bråten, A. Jørgensen, R. Wiseth and L. R. Hellevik, 'Machine learning augmented reduced-order models for ffr-prediction,' *Computer Methods in Applied Mechanics and Engineering*, vol. 384, p. 113 892, 2021, ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2021.113892. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045782521002292.

[18] J. H. Medicine, *Atherosclerosis*. [Online]. Available: https://www.hopkinsmedicine.org/health/conditions-and-diseases/atherosclerosis (visited on 06/02/2023).

[19] E. Hulten and D. W. Carlson, in *Atherosclerosis: Clinical Perspectives Through Imaging*, A. J. Taylor and T. C. Villines, Eds. London: Springer London, 2013, ISBN: 978-1-4471-4288-1. DOI: `10.1007/978-1-4471-4288-1_6`. [Online]. Available: `https://doi.org/10.1007/978-1-4471-4288-1_6`.

[20] N. H. Pijls, W. F. Fearon, P. A. Tonino, U. Siebert, F. Ikeno, B. Bornschein, M. van't Veer, V. Klauss, G. Manoharan, T. Engstrøm *et al.*, 'Fractional flow reserve: The fame (fractional flow reserve versus angiography for multivessel evaluation) trial,' *Journal of the American College of Cardiology*, vol. 55, no. 3, pp. 281–285, 2010.

[21] *Diagnostic Radiology Physics* (Non-serial Publications). Vienna: INTERNATIONAL ATOMIC ENERGY AGENCY, 2014, ISBN: 978-92-0-131010-1. [Online]. Available: `https://www.iaea.org/publications/8841/diagnostic-radiology-physics`.

[22] A. C. Kwan, G. Cater, J. Vargas and D. A. Bluemke, 'Beyond coronary stenosis: Coronary computed tomographic angiography for the assessment of atherosclerotic plaque burden,' *Current Cardiovascular Imaging Reports*, vol. 6, no. 2, pp. 89–101, 2013. DOI: `10.1007/s12410-012-9183-z`.

[23] P. Carrascosa, J. A. Leipsic, C. Capunay, A. Deviggiano, J. Vallejos, A. Goldsmit and G. A. Rodriguez-Granillo, 'Monochromatic image reconstruction by dual energy imaging allows half iodine load computed tomography coronary angiography,' *European Journal of Radiology*, vol. 84, no. 10, pp. 1915–1920, 2015. DOI: `10.1016/j.ejrad.2015.06.019`.

[24] R. Szeliski, *Computer vision: Algorithms and applications*. Springer, 2022.

[25] V. Lakshmanan, M. Görner and R. Gillard, *Practical machine learning for computer vision: End-to-end machine learning for images*. O'Reilly, 2021.

[26] L. Ghiani, A. Sassu, F. Palumbo, L. Mercenaro and F. Gambella, 'In-field automatic detection of grape bunches under a totally uncontrolled environment,' *Sensors*, vol. 21, Jun. 2021. DOI: `10.3390/s21113908`.

[27] X. Luo, W. Liao, J. Xiao, J. Chen, T. Song, X. Zhang, K. Li, D. N. Metaxas, G. Wang, S. Zhang and et al., 'Word: A large scale dataset, benchmark and clinical applicable study for abdominal organ segmentation from ct image,' *Medical Image Analysis*, vol. 82, p. 102 642, 2022. DOI: `10.1016/j.media.2022.102642`.

[28] L. R. Dice, 'Measures of the amount of ecologic association between species,' *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.

[29] A. A. Taha and A. Hanbury, 'Metrics for evaluating 3d medical image segmentation: Analysis, selection, and tool,' *BMC Medical Imaging*, vol. 15, no. 1, p. 29, 2015.

[30] D. P. Huttenlocher, G. A. Klanderman and W. J. Rucklidge, 'A multi-resolution technique for comparing images using the hausdorff distance,' *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 9, pp. 850–863, 1993.

[31] F. Milletari, N. Navab and S.-A. Ahmadi, 'V-net: Fully convolutional neural networks for volumetric medical image segmentation,' in *2016 Fourth International Conference on 3D Vision (3DV)*, 2016, pp. 565–571.

[32] T.-Y. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, 'Focal loss for dense object detection,' in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[33] K. He, X. Chen, S. Xie, Y. Li, P. Dollár and R. Girshick, *Masked autoencoders are scalable vision learners*, 2021. DOI: 10.48550/ARXIV.2111.06377. [Online]. Available: https://arxiv.org/abs/2111.06377.

[34] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura and R. M. Summers, 'Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning,' *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[35] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, 'Imagenet large scale visual recognition challenge,' *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[36] O. Ronneberger, P. Fischer and T. Brox, 'U-net: Convolutional networks for biomedical image segmentation,' *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, 2015.

[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser and I. Polosukhin, *Attention is all you need*, 2017. DOI: 10.48550/ARXIV.1706.03762. [Online]. Available: https://arxiv.org/abs/1706.03762.

[38] *Language modelling*. [Online]. Available: https://paperswithcode.com/task/language-modelling (visited on 20/02/2023).

[39] A. FUJII, *Do vision transformers see like convolutional neural networks? (paper explained)*, Oct. 2021. [Online]. Available: https://towardsdatascience.com/do-vision-transformers-see-like-convolutional-neural-networks-paper-explained-91b4bd5185c8 (visited on 18/01/2023).

[40] H. Bao, L. Dong, S. Piao and F. Wei, *Beit: Bert pre-training of image transformers*, 2021. DOI: 10.48550/ARXIV.2106.08254. [Online]. Available: https://arxiv.org/abs/2106.08254.

[41] Z. Xie, Z. Zhang, Y. Cao, Y. Lin, J. Bao, Z. Yao, Q. Dai and H. Hu, *Simmim: A simple framework for masked image modeling*, 2021. DOI: 10.48550/ARXIV.2111.09886. [Online]. Available: https://arxiv.org/abs/2111.09886.

[42] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan and S. Belongie, *Feature pyramid networks for object detection*, 2016. DOI: `10.48550/ARXIV.1612.03144`. [Online]. Available: `https://arxiv.org/abs/1612.03144`.

[43] Y. Tang, D. Yang, W. Li, H. R. Roth, B. Landman, D. Xu, V. Nath and A. Hatamizadeh, 'Self-supervised pre-training of swin transformers for 3d medical image analysis,' *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. DOI: `10.1109/cvpr52688.2022.02007`.

[44] P. Soille, *Morphological Image Analysis: Principles and Applications*. Springer, 2003.

[45] P. Medrano-Gracia, J. Ormiston, M. Webster, S. Beier, C. Ellis, C. Wang, A. A. Young and B. R. Cowan, 'Construction of a coronary artery atlas from ct angiography,' in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014*, P. Golland, N. Hata, C. Barillot, J. Hornegger and R. Howe, Eds., Cham: Springer International Publishing, 2014, pp. 513–520, ISBN: 978-3-319-10470-6.

[46] Y. Wang, X. Wei, F. Liu, J. Chen, Y. Zhou, W. Shen, E. K. Fishman and A. L. Yuille, 'Deep distance transform for tubular structure segmentation in ct scans,' in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3833–3842.

[47] A. F. Frangi, W. J. Niessen, K. L. Vincken and M. A. Viergever, 'Multiscale vessel enhancement filtering,' in *Medical Image Computing and Computer-Assisted Intervention—MICCAI'98: First International Conference Cambridge, MA, USA, October 11–13, 1998 Proceedings 1*, Springer, 1998, pp. 130–137.

[48] L. Antiga, M. Piccinelli, L. Botti, B. Ene-Iordache, A. Remuzzi and D. A. Steinman, 'An image-based modeling framework for patient-specific computational hemodynamics,' *Medical & Biological Engineering & Computing*, vol. 46, no. 11, pp. 1097–1112, 2008.

[49] L. Antiga, B. Ene-Iordache and A. Remuzzi, 'Centerline computation and geometric analysis of branching tubular surfaces with application to blood vessel modeling,' Feb. 2003.

[50] N. P. Johnson, D. T. Johnson, R. L. Kirkeeide, C. Berry, B. De Bruyne, W. F. Fearon, K. G. Oldroyd, N. H. Pijls and K. L. Gould, 'Repeatability of fractional flow reserve despite variations in systemic and coronary hemodynamics,' *JACC: Cardiovascular Interventions*, vol. 8, no. 8, pp. 1018–1027, 2015. DOI: `10.1016/j.jcin.2015.01.039`.

[51] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, 'Pytorch: An imperative style, high-performance deep learning library,' in *Advances in Neural Information Processing Systems*, 2019, pp. 8024–8035.

[52] W. Falcon, S. Dong, J. Bourgeois, N. Keskar, S. Chintala and E. Larysa, 'Pytorch lightning: A lightweight pytorch wrapper for ml researchers,' in *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[53] *Monai: Medical open network for ai*, `https://monai.io`, 2020. (visited on 10/02/2023).

[54] A. Diaz-Pinto, S. Alle, V. Nath, Y. Tang, A. Ihsani, M. Asad, F. Pérez-García, P. Mehta, W. Li, M. Flores, H. R. Roth, T. Vercauteren, D. Xu, P. Dogra, S. Ourselin, A. Feng and M. J. Cardoso, *Monai label: A framework for ai-assisted interactive labeling of 3d medical images*, 2023. arXiv: `2203.12362 [cs.HC]`.

[55] A. Fedorov, R. Beichel, J. Kalpathy-Cramer, J. Finet, J.-C. Fillion-Robin, S. Pujol, C. Bauer, D. Jennings, F. Fennessy, M. Sonka *et al.*, '3d slicer as an image computing platform for the quantitative imaging network,' *Magnetic resonance imaging*, vol. 30, no. 9, pp. 1323–1341, 2012.

[56] R. Kikinis, S. D. Pieper and K. G. Vosburgh, '3d slicer: A platform for subject-specific image analysis, visualization, and clinical support,' *Intraoperative Imaging*, vol. 3, pp. 277–289, 2014.

[57] S. Pieper, B. Lorensen, W. Schroeder and R. Kikinis, 'The na-mic kit: Itk, vtk, pipelines, grids and 3d slicer as an open platform for the medical image computing community,' in *Biomedical Imaging: Nano to Macro, 2004. IEEE International Symposium on*, IEEE, 2004, pp. 698–701.

[58] *Medical image segmentation on synapse multi-organ ct*. [Online]. Available: `https://paperswithcode.com/sota/medical-image-segmentation-on-synapse-multi` (visited on 20/02/2023).

[59] Project-MONAI, *Research-contributions/swinunetr/btcv at main · project-monai/research-contributions*. [Online]. Available: `https://github.com/Project-MONAI/research-contributions/tree/main/SwinUNETR/BTCV` (visited on 15/03/2023).

[60] MIC-DKFZ, *Mic-dkfz/nnunet*. [Online]. Available: `https://github.com/MIC-DKFZ/nnUNet/tree/nnunetv1` (visited on 05/03/2023).

[61] *Asoca grand challenge leaderboard*. [Online]. Available: `https://asoca.grand-challenge.org/evaluation/challenge/leaderboard/` (visited on 02/05/2023).

Appendix A

Clinical Characteristics for St. Olavs Hospital Dataset

Additional clinical characteristics for the St. Olavs Hospital dataset.

Table A.1: Clinical characteristics

Variables/ clinical characteristics	Values
Age years (mean ±SD)	61,8(±8,00)
Male n(%)	76(67)
Female n(%)	38(33)
BMI (kg/m2, mean±SD)	27,5(±3,5)
Heart rate	67(±12,6)
Smoking/ former smoking	58(51)
Hypertension	57(50)
Dyslipidemia	50(44)
Family predisposition	38(33)
Previous CAD events	1(1.0)
Diabetes	16(14)
Clinical classification of angina	
Non angina chest pain	16(14)
Atypical	54(47)
Typical	44(39)
TIA/stroke	10(9)
COPD/asthma	2(2)
Atrial fibrillation	8(7)
Time from CCTA to ICA	35(±15)
Effective CCTA radiation dose: 0,014	2,79(±3,16) Median 2,08
Effective CCTA radiation dose: 0,026	5,19(±3,88)
Pretest probability, ESC 2013	55,5(±20,0)
Pretest probability, ESC 2019	23,4(±11,68)
β-blocker	24(21)
Calsium antagonists	14(12)
Nitrates	19(17)
ACE inhibitor or ARB	39(34)
Aspirin/another platelet inhibitor	97(85)
Statin/another lipid-modifying agent	98(86)

Appendix B

Asoca Grand Challenge leaderbord

A snapshot of the online leaderboard for the ASOCA Grand Challenge, with our submission [61]. The snapshot was taken at the 5th of May 2023.

Entry	User	Created	Dice Coefficient	95 Hausdorff Distance
1st	liruikun	26 March 2021	**0.8856**	3.4128
2nd	junma	23 Sept. 2020	0.88	2.6875
3rd	shenl19	22 Sept. 2020	0.8794	4.4087
4th	bigPYJ	21 Sept. 2020	0.8736	3.383
5th	RuochenGao	3 Oct. 2020	0.8711	2.1455
6th	shine12100	20 Sept. 2020	0.8675	4.0973
7th	**michael.larsen90** *	2 May 2023	0.8663	5.7943
8th	erebos1122	22 Sept. 2020	0.8653	4.0609
9th	aaa15643	20 Sept. 2020	0.8603	3.8238
10th	ys810137152	21 Sept. 2020	0.8537	**1.8115**
11th	q56101044	15 Dec. 2022	0.8512	8.2606
12th	xf4j	23 Sept. 2020	0.8382	8.5471
13th	LiangLab	2 May 2023	0.8228	7.3533
14th	brunom0liveira91	17 Jan. 2022	0.8067	3.648
15th	hyt	22 Sept. 2020	0.8033	4.613
16th	nanwaychen	3 Oct. 2020	0.799	6.0894
17th	Gpeppa	24 April 2023	0.789	34.2364
18th	raahus	21 Sept. 2020	0.7813	4.7718
19th	jnk50	2 May 2023	0.7478	7.2996
20th	zohaib1122	21 Sept. 2020	0.6904	8.0434
21st	Shisheng	22 July 2022	0.5924	27.1515
22nd	1319084952	12 Feb. 2023	0.0412	44.3244
23rd	xin.zhang.3	22 Sept. 2020	0.0399	103.4254
24th	hongqq	4 April 2023	0.0211	81.4007
25th	palkia	29 Sept. 2020	0.0005	71.1053

Table B.1: Results from ASOCA Grand challenge online leaderboard. * denotes our submission

Source: [61]

NTNU
Norwegian University of
Science and Technology