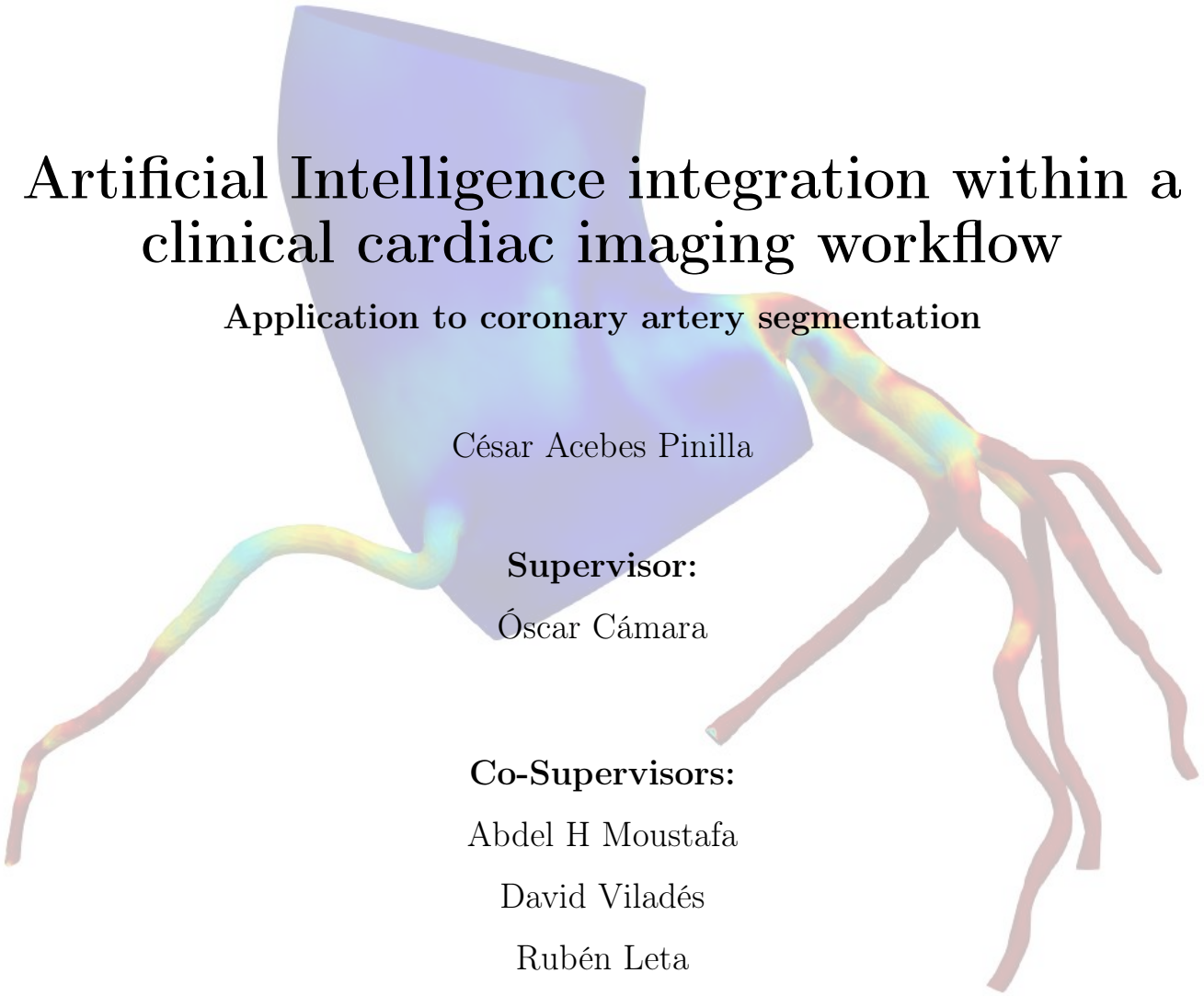
Master thesis on Computational Biomedical Engineering

Universitat Pompeu Fabra



Artificial Intelligence integration within a clinical cardiac imaging workflow

Application to coronary artery segmentation

César Acebes Pinilla

Supervisor:

Óscar Cámara

Co-Supervisors:

Abdel H Moustafa

David Viladés

Rubén Leta

July 2021

upf. Universitat Pompeu Fabra Barcelona

HOSPITAL DE LA SANTA CREU I SANT PAU

UNIVERSITAT AUTÒNOMA DE BARCELONA

Master thesis on Computational Biomedical Engineering

Universitat Pompeu Fabra

Artificial Intelligence integration within a clinical cardiac imaging workflow

Application to coronary artery segmentation

César Acebes Pinilla

Supervisor:

Óscar Cámara

Co-Supervisors:

Abdel H Moustafa

David Viladés

Rubén Leta

July 2021

Contents

1 Introduction **1**

1.1 Artificial Intelligence (AI) in clinical environments 1

1.1.1 Regular workflow in a Cardiac Image Unit 1

1.1.2 Importance of AI in clinical workflows 2

1.1.3 Medical data processing 4

1.1.4 Segmentation of medical images 5

1.2 Use case: Coronary arteries segmentation 7

1.2.1 Anatomy of the coronary arteries 8

1.2.2 Coronary artery disease 9

1.3 Artificial Intelligence in CAD 10

1.4 Objectives 13

2 Methods **15**

2.1 General workflow 15

2.2 Description of workflow modules 16

2.2.1 Computational module 16

2.2.2 Visualization and post-processing module 18

2.2.3 Statistical analysis module 20

2.2.4 External modules 20

2.2.5 Clinical module 20

2.3 Use case: Coronary artery segmentation 22

2.3.1 Coronary dataset creation 22

2.3.2 Design of the segmentation model 27
2.3.3 Final segmentation model 33
2.3.4 Deployment of the model 34
2.3.5 Proof-of-concept workflow 35

3 Results **37**

3.1 Segmentation model 37
3.1.1 Dataset creation and first training 37
3.1.2 Design of the segmentation model 40
3.1.3 Final model 45
3.2 Proof-of-concept 46
3.2.1 Mesh generation and post-processing 46
3.2.2 Visualization of the results 48

4 Discussion **50**

4.1 Coronary segmentation model 50
4.2 Data management and deployment 52
4.3 Other potential applications 53

5 Conclusions **55**

List of Figures **57**

List of Tables **60**

Bibliography **61**

A First Appendix **69**

A.1 Analysis of the first segmentation algorithm. 69

B Second Appendix **75**

B.1 Workflow proof-of-concept 75

B.1.1 Segmentation and mesh generation 75
B.1.2 Mesh post-processing 75
B.1.3 Computational Fluid Dynamics simulation 77
B.1.4 Visualization of the results 81

Any sufficiently advanced technology is indistinguishable from magic.

Arthur C. Clarke

Acknowledgement

First of all, I would like to take this opportunity to warmly thank my tutor, Óscar Cámara, for giving me the opportunity to carry out this fantastic project. I would also like to highlight his support during the process and his efficient help; as well as to the other members of the Physense group, in particular Mayra Morello, for the incredible support during the endless hours of work at the Hospital and for her help with the CFD simulations.

In addition, for the daily support during the whole project, his energy, and all the teachings I have received from him: I would especially like to thank Abdel Moustafa. Without him, this project would not have been possible. I would also like to extend my sincere gratitude to the rest of the members of the Cardiac Imaging Unit of the *Hospital de Sant Pau*, especially David Viladés for his priceless clinical feedback and insight and Rubén Leta for allowing me to join the unit and sharing with me his extensive knowledge in the field of cardiac imaging.

Finally, my family and friends have been there all along, supporting me, and I really must thank them.

Abstract

Coronary Artery Disease (CAD) has been the leading cause of death in industrialized countries since the 1960s. It is a multi-factorial disease consisting of plaque accumulation in the coronary vessels, leading to stenosis or vessel narrowing, or plaque rupture. The combination of these factors is the primary etiology of angina, myocardial infarct, and sudden death. The analysis of stenosis degree and plaque composition is carried out by computed tomography (CT) scanners. For precise quantification, it is necessary to process cardiac CT (CCT) images automatically, including the coronary arterial lumen segmentation and the computation of disease-related metrics. In addition, computational fluid simulations (CFD) could also be carried out to assess hemodynamic factors non-invasively. To this end, the development of Artificial Intelligence (AI)-based algorithms has proven their accuracy in medical imaging segmentation tasks. However, despite their good performance, it is difficult to deploy them in clinical environments due to poor adaptation of health care facilities to the new AI. This way, this thesis aims to develop an AI-based workflow that enables deep learning-based algorithms to process multi-modality image data in cardiology in a clinical environment. The segmentation of coronary vessels from CCT images was taken as an use case, and a proof-of-concept was carried out.

The workflow consists of a series of modules through which the images are processed. First, they passed through a computational module that includes the segmentation algorithms. To this end, the nnU-Net framework, allowing the use of U-Net architectures and their automatic hyperparameter selection, was used. Some experiments were designed and set up to determine the proper steps to achieve an optimal segmentation. Based on feedback from those experiments, a final segmentation model was created. Furthermore, the proof-of-concept analyzes the suitability of using the workflow for coronary arterial lumen segmentation and posterior CFD simulations. Regarding the deployment in a hospital, several options were analyzed.

The results show that the proposed methodology can correctly segment most coronaries on CCT images, with similar results in time and performance to available

commercial software and used in cardiac imaging units nowadays. Regarding the results of CFD simulations, they agree with physiological values.

The workflow proposed in this thesis represents a step forward in developing clinical workflows as it facilitates the integration of different image processing and analysis modules. Furthermore, if a training dataset including a more representative number of samples were used, the performance of the algorithm could be substantially improved, thus being ready for implementation.

Keywords: Artificial Intelligence; Machine learning; Segmentation; Cardiovascular Disease; Cardiac Imaging.

Chapter 1

Introduction

1.1 Artificial Intelligence (AI) in clinical environments

1.1.1 Regular workflow in a Cardiac Image Unit

The Radiology department in a hospital is the facility where the equipment and professionals who provide radiological images of the human body's interior are located. These images can be obtained through various physical agents, including X-rays, magnetic fields, or ultrasound, among others. It can be considered one of the most critical departments of the hospital, as it is where images are acquired that are used to make the prediction, diagnosis, and follow-up of diseases that are otherwise difficult to assess.In a radiology department, the usual workflow for a patient is the following: first, the patient goes to the hospital, and the physician recommends that he or she undergoes a particular radiological study, selecting the most appropriate imaging modality; after that, the study is performed, and the images are stored in the PACS of the hospital. This acronym stands for *Picture Archiving and Communication System* and refers to a system for storing, distributing, and viewing medical images obtained from multiple modalities. Then, once the images are obtained, they are post-processed and analyzed by the corresponding expert to make a diagnosis.

In some exceptional cases, additional imaging studies may be requested.

One of the clinical areas in which radiologic diagnostic imaging is fundamental is Cardiology since the heart is an internal structure of the body, and it is necessary to evaluate it in a non-invasive way. To this end, many hospitals have a unit dedicated exclusively to cardiac imaging. Echocardiography, cardiac magnetic resonance, computerized tomography (CT), and nuclear medicine are some of the imaging techniques used to acquire images for further evaluation of the physical and functional state of the heart. After obtaining these images, the next step is their processing and interpretation by a cardiac imaging expert and further evaluation. Depending on the imaging modality and the heart structure, the processing is carried out differently, tending to be manual and labor-intensive. There, an important step is the segmentation of the cardiac images, which consists of a partition of the image into a set of relevant regions corresponding to different heart areas [1]. In cardiac imaging, an example of segmentation could be the semantic division of the heart into the four different chambers and the myocardial wall.

Due to the large number of images to be processed manually and the limited time dedicated to a single patient (around 20 minutes), this step has historically represented a bottleneck in cardiac imaging processing. To overcome this limitation, some machine learning (ML) techniques have been initially used to perform cardiac segmentation, and registration tasks [2]. In addition, more recently, the development of Deep Learning (DL) has made a significant contribution to the automation of these tasks. Thus, not only has the development of state-of-the-art algorithms been fundamental, but the ease of access to large annotated databases or advanced computational infrastructures has also transformed the field of medical image processing. Hence, DL and ML are part of the so-called AI, and both can be included in the AI revolution of recent years.

1.1.2 Importance of AI in clinical workflows

AI is an interdisciplinary branch of computer science that allows computers to adopt human-like behaviors, such as generalizing, learning patterns, or acquiring knowl-

edge from past experiences. It enables, among other things, to work in problem-solving tasks by classifying unseen examples or predicting results without knowing the parameters of a model. In this respect, the introduction of AI algorithms into clinical environments has enabled to deal with a significant amount of vast data. Moreover, it can help clinicians in otherwise time-consuming image analysis tasks, as they can identify imperceptible patterns for human reasoning embedded in large amounts of seemingly unrelated data [3].

Over the past few years, examples of AI in clinical environments have grown exponentially. Some of them are optimizing surgical processes, enabling a real-time evaluation of them; patient screening, allowing a more precise and immediate description of their condition; or unsupervised hierarchical clustering of patients. Some concrete examples in each step of the cardiac imaging workflow are presented in Figure 1.

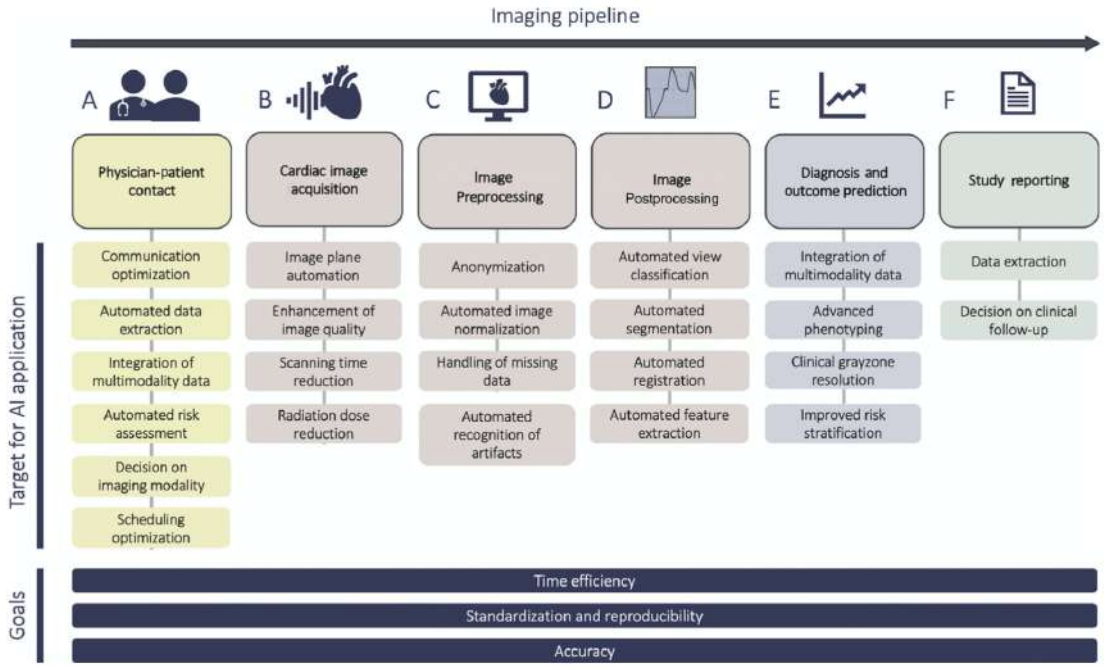


Figure 1: Pipeline showing the steps of the cardiac imaging workflow. From Loncaric, F., et al. [4].

However, proper implementation of these innovative AI workflows in clinical routine is not trivial; it presents several challenges, some of the most important of which are presented in this section, using data from [5], [6], [7] and [8].

In the first place, it is essential to consider the clinical utility of the AI algorithms since a successful validation of the models does not necessarily guarantee clinical utility. Related to it, the generalization of a model to new populations poses a problem. It may incur some bias if these AI models are trained with data provided in Challenges, such as ASOCA[1] or EMIDEC [2], which generally are well-curated, and tested with real-world clinical data from everyday patients, i.e., often incomplete, heterogeneous and of low-quality. In addition, it is critical to keep in mind that these models are not static over time and that the more data samples are available, the more necessary model re-training will be, requiring a new set of supervised labeled data each time this re-training is undertaken.

In addition, there is a need for a user-friendly workflow for physicians to indicate how they or their patients can benefit from the algorithms, as there are a large number of scientific publications that do not include such information. Suppose you finally agree to implement this workflow in the clinical routine. In that case, other considerations should be taken into account, including the logistical difficulties caused by the poor adaptation of health care facilities to the new AI, such as the complexity and high cost of data storage. Data labeling is another point to consider since it is an indispensable requirement to develop solid models. If this labeling is not carried out under a standard protocol, labeling errors can make it challenging to train the algorithms correctly. Finally, the human factor can be misleading if the information provided by the model is not used in the right way by clinicians, with harmful consequences for the patient.

1.1.3 Medical data processing

Despite all of the above challenges, the proper implementation of an AI workflow in cardiac imaging should be beneficial primarily to the patients in many aspects, some of which can be seen in Figure 2. A typical pipeline of a process involving cardiac imaging data is displayed, presenting several steps in which the introduction of AI could imply a new paradigm. For example, the acquisition of images could

[1] https://asoca.grand-challenge.org/
[2] http://emidec.com/

be faster and their quality higher, facilitating their incorporation into an automatic segmentation algorithm. In this way, the necessary computational markers of disease could be obtained for the cardiologist to evaluate the condition. This diagnosis could even be made through AI, combining the above results with demographic, genetic, and other data in a short time and under the supervision of the physician. In addition to disease diagnostic, medical image processing also allow conducting patient follow-up and prediction of illnesses [9]. Regarding multidimensional imaging processing, some of the most important applications for AI have been image enhancement [10], image-based diagnosis [11], or segmentation of structures of interest [12] for multiple organs, such as brain or heart, or tumors.

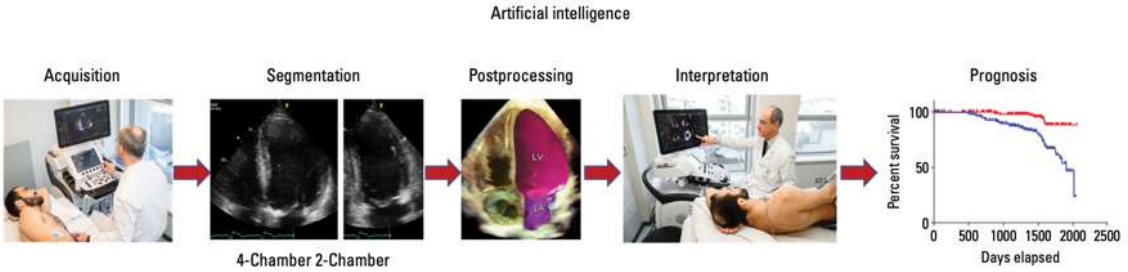


Figure 2: Artificial intelligence impact in the various steps of the cardiac imaging pipeline. Adapted from Badano, L. et al. [3].

1.1.4 Segmentation of medical images

Among these applications, the automatic segmentation of structures stands out due to the usefulness of knowing precisely the location and limits of the different structures of the human body in an unsupervised way, allowing for anatomical, functional and quantitative assessment. Its great importance and interest have been such that its research accounts for 70% of the international challenges for image analysis in bio-medicine [13]. It means that the number and complexity of the algorithms aimed to compute segmentations have notably increased from noisy but fast region-, edge- and threshold-based methods to slow but very accurate artificial neural network-based ones [14].

In the recent years, the latter have become the gold standard in medical image segmentation, with convolutional neural networks (CNNs) tailored to these tasks being the most important architectures. The introduction of the U-Net [15] in 2015 has been one of the most important breakthroughs in this field, allowing training the segmentation algorithm with fewer training images and being considered the most successful architecture in the medical domain [16]. The U-Net is based on CNNs and presents an encoder-decoder architecture, consisting of a contraction path to capture the context of the image and a symmetrical expanding path that enables precise localization, as shown in Figure 3. Due to its high accuracy in medical tasks, U-Net, in combination with other features, constitutes most of the most successful algorithms for 3D segmentation, even more, if the pipeline is appropriately designed and the hyper-parameters of the model are correctly selected.

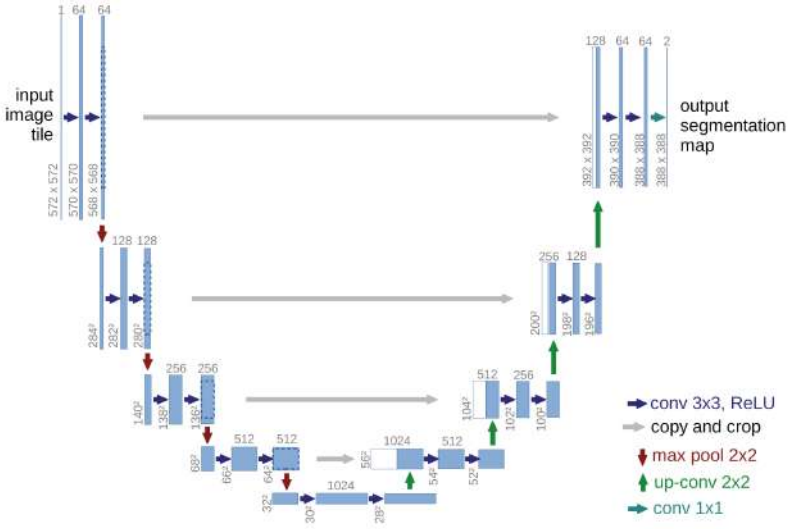
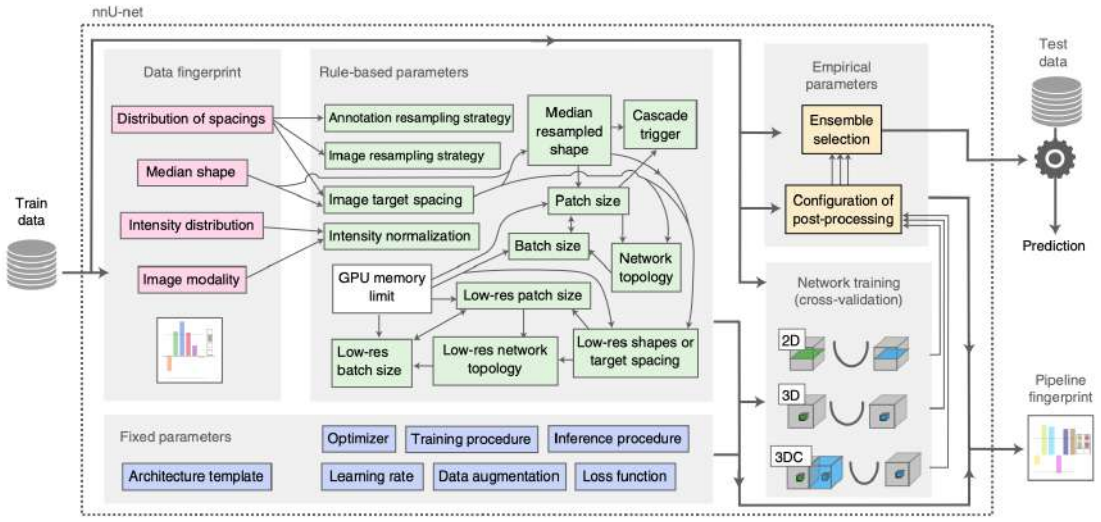


Figure 3: Example U-net architecture. From Ronneberger, O. et al. [15].

More recently, a new segmentation framework, the nnU-Net (no new net, Isensee et al., 2020), has become the most popular choice for 3D medical image segmentation tasks [17]. It is a dynamically adaptive framework built directly around the original U-Net architecture and allowing automatic selection of network configuration and hyper-parameters, as it can be seen in Figure 4. In this sense, the automation of manual steps that it enables, its network engineering and processing capabilities, coupled with its flexibility and power in an open-source software, have brought the framework to the forefront of the latest medical image segmentation challenges [18].



Design choice	Required input	Automated (fixed, rule-based or empirical) configuration derived by distilling expert knowledge (more details in online methods)
Learning rate	–	Poly learning rate schedule (initial, 0.01)
Loss function	–	Dice and cross-entropy
Architecture template	–	Encoder–decoder with skip-connection ('U-Net-like') and instance normalization, leaky ReLU, deep supervision (topology-adapted in inferred parameters)
Optimizer	–	SGD with Nesterov momentum ($\mu = 0.99$)
Data augmentation	–	Rotations, scaling, Gaussian noise, Gaussian blur, brightness, contrast, simulation of low resolution, gamma correction and mirroring
Training procedure	–	1,000 epochs × 250 minibatches, foreground oversampling
Inference procedure	–	Sliding window with half-patch size overlap, Gaussian patch center weighting
Intensity normalization	Modality, intensity distribution	If CT, global dataset percentile clipping & z score with global foreground mean and s.d. Otherwise, z score with per image mean and s.d.
Image resampling strategy	Distribution of spacings	If anisotropic, in-plane with third-order spline, out-of-plane with nearest neighbor Otherwise, third-order spline
Annotation resampling strategy	Distribution of spacings	Convert to one-hot encoding → If anisotropic, in-plane with linear interpolation, out-of-plane with nearest neighbor Otherwise, linear interpolation

Image target spacing	Distribution of spacings	If anisotropic, lowest resolution axis tenth percentile, other axes median. Otherwise, median spacing for each axis. (computed based on spacings found in training cases)
Network topology, patch size, batch size	Median resampled shape, target spacing, GPU memory limit	Initialize the patch size to median image shape and iteratively reduce it while adapting the network topology accordingly until the network can be trained with a batch size of at least 2 given GPU memory constraints. for details see online methods.
Trigger of 3D U-Net cascade	Median resampled image size, patch size	Yes, if patch size of the 3D full resolution U-Net covers less than 12.5% of the median resampled image shape
Configuration of low-resolution 3D U-Net	Low-res target spacing or image shapes, GPU memory limit	Iteratively increase target spacing while reconfiguring patch size, network topology and batch size (as described above) until the configured patch size covers 25% of the median image shape. For details, see online methods.
Configuration of post-processing	Full set of training data and annotations	Treating all foreground classes as one; does all-but-largest-component-suppression increase cross-validation performance? Yes, apply; reiterate for individual classes No, do not apply; reiterate for individual foreground classes
Ensemble selection	Full set of training data and annotations	From 2D U-Net, 3D U-Net or 3D cascade, choose the best model (or combination of two) according to cross-validation performance

Figure 4: Representation of the nnU-net configuration and description of the hyperparameters. From Isensee, F. et al. [17].

1.2 Use case: Coronary arteries segmentation

In this thesis, a use case has been proposed and developed, which is the segmentation of the coronary arterial lumen from CCT images. This section aims to establish a theoretical framework to understand the pathophysiology behind this application.

1.2.1 Anatomy of the coronary arteries

The heart is the center of the cardiovascular system and is responsible for pumping blood uninterruptedly throughout our lives. Given its importance, a constant supply of oxygen and nutrient-rich blood is essential to the heart to achieve its goal efficiently. Thus, blood supply to the cardiac muscle is achieved thanks to an efficient network of blood vessels, called the coronary tree, which is wrapped around the heart.

It consists of two main coronary arteries (left and right ones), originating from the left and right cusps of the aortic valve. The left coronary artery is bifurcated into the left anterior descending and the circumflex arteries, perfusing most of the left ventricle. The right coronary artery ramifies into the right marginal arteries and, in most people, into the posterior descending artery. The latter is responsible for perfusing the right ventricle and the inferior segments of the left ventricular myocardium. Figure 5 display the main coronary vessels.

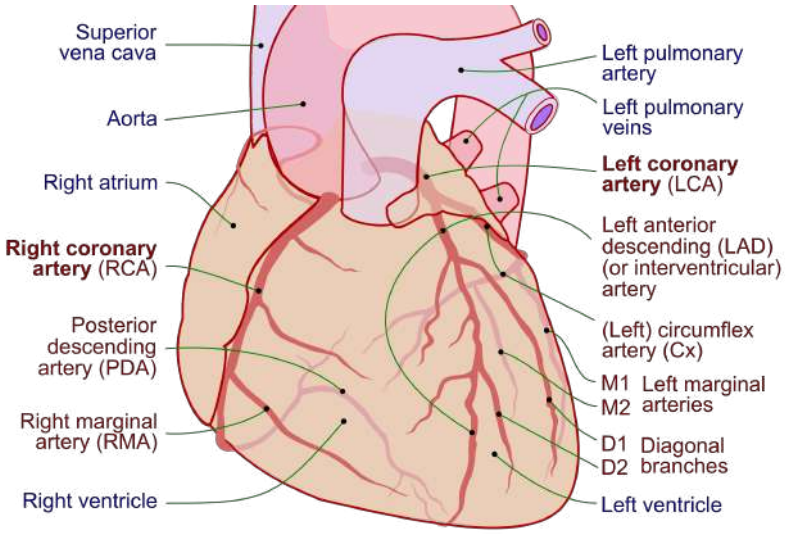


Figure 5: Anatomy of the coronary tree, with coronary arteries labeled in red text and other landmarks in blue text. From Lynch, P. J. [19].

However, the organization and arrangement of the vessels pose a severe problem in the event of an obstruction. If there is a decrease in flow in some of them, the lack of interconnection between them does not allow the blood flow to be diverted, producing ischemia in the cardiac muscular tissue irrigated by the affected vessel.

If this decrease in flow has been produced by accumulated cholesterol and fibrous tissue, it can break down and produce a blockage in the arteries, with the consequent death of the surrounding cardiac muscle tissue. This process is called myocardial infarct and requires immediate treatment to survive.

1.2.2 Coronary artery disease

Etiology and importance of coronary artery disease

Myocardial infarct and ischemia are two of the most potentially dangerous consequences of coronary artery disease (CAD), which is a multifactorial disease that develops when there exists an increased blood concentration of low-density lipoprotein (LDL) [20] or at lower levels of it combined with other risk factors, such as glucose intolerance, isolated systolic hypertension, or smoking, among others [21] [22].

When the diet is rich in LDL, these lipoproteins tend to accumulate and aggregate in the sites with a lower or oscillatory shear stress of the tunica intima, i.e., the innermost layer of the arteries, inducing the migration of monocytes from the bloodstream into the intima [21], due to the potential menace that LDL suppose. Once there, the mononuclear cells differentiate into macrophages, absorb the lipoproteins, and form foam cells, which will develop the necrotic core of the lesion after their apoptosis. This core will increase in size after the secretion of fibrous aggregates by smooth muscle cells, originated from the tunica media and grow to the *tunica adventitia* until arriving at a point when the lesion begins to expand in the direction of the lumen of the vessel [23].

Plaque accumulation results in remodeling of the lumen, which can lead to stenosis if plaque growth is continued or plaque rupture. These factors, combined, are the primary etiology of angina, myocardial infarct, and sudden death [24]. However, the severity of the lesion depends on the composition of the plaque; since soft plaques, mainly composed of lipids, are more vulnerable and prone to cause thrombi.

CAD has been the leading cause of death in industrialized countries since the 1960s due to increased average life expectancy and population growth [25]. Since then, in developing countries, there has been a progressive increase in the incidence of CAD, produced by the globalization of the western diet, and a rise in sedentary behavior [26] [27] that has led to it being the leading cause of death around the world [28].

Cardiac imaging in coronary disease

To evaluate the stenotic problem in a non-invasive way, the development of computed tomography angiography (CTA) has been an important step, allowing the introduction of coronary CTA, being an alternative to the traditional invasive coronary angiography, which is associated with the risk of serious complications [29][30]. Traditionally, measuring the degree of stenosis has been employed for diagnosing the severity of the disease (utilizing the fractional flow reserve or FFR) [31]. However, this is not sufficient for a complete assessment of the risk of disease, especially when the plaque development is in the first stages [24]. In this regard, modern CT scanners allow the visualization of the coronary lumen, in addition to the vessel wall, of the plaque burden, and of the perivascular adipose tissue, highly related to CAD [30][32][33]. Figure 6 highlights the utility of cardiac CT (CCT) in assessing CAD. On the one hand, the lumen is a visible and clear structure in contrasted CCT images; however, the plaque is a more complex structure to segment since its composition can vary, depending on its calcium content. Thereby, the plaques are divided into calcified, non-calcified, and mixed calcified, presenting different radiodensity properties, among others [34].

1.3 Artificial Intelligence in CAD

In recent years, the number of publicly available datasets for segmentation has increased, as has the number of scientific articles for DL-based cardiac image segmentation. Among the latter, there are those based on delineation of the coronary arterial lumen [35] and the ones aimed at segmenting calcified and non-calcified coronary plaque [1], both of them necessary for diagnosing CAD.

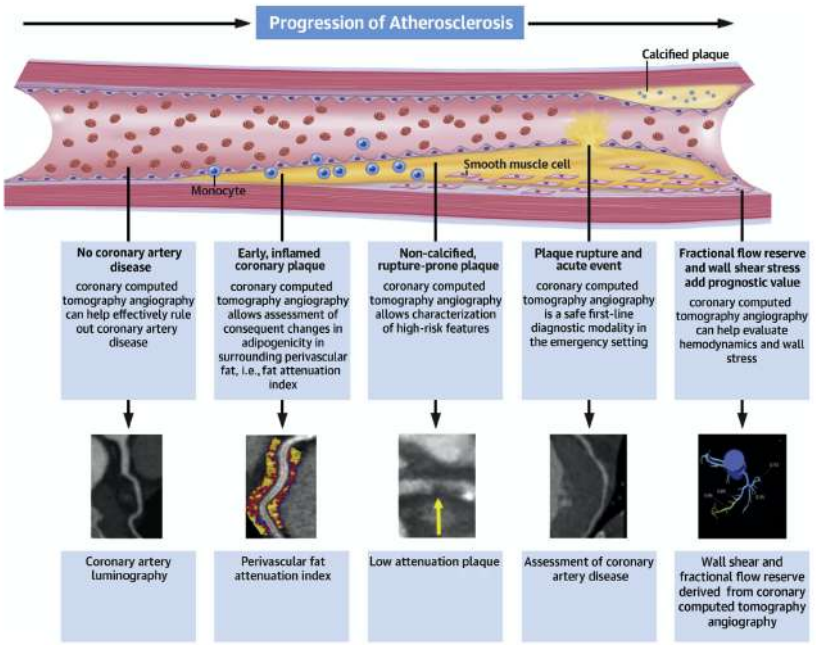


Figure 6: Utility of cardiac computed tomography in coronary artery disease. From Abdelrahman, K.M. et al. [33].

First of all, coronary arterial lumen segmentation would allow for detecting and quantifying possible stenosis, performing computational fluid dynamics (CFD) studies, or creating anatomical models, among other applications. For instance, the severity of the stenoses is quantified nowadays employing the FFR, an invasive measurement consisting of introducing a catheter in the coronary artery under hyperemic conditions. An automatic lumen segmentation would allow extracting the vessel's surface, thus computing the FFR in a non-invasive manner [36] [37]. The latter is commercially available offered by Heartflow[3], which offers a cloud-based platform of super-computing nodes for its computation. Regarding CFD, having a segmentation for the coronaries and the ascending portion of the aorta is the first step to developing a volumetric mesh. In that mesh, CFD simulations can be computed to export hemodynamic factors related to atherosclerosis [38] such as the wall shear stress (WSS), as in [39]. Finally, the anatomy of the coronary lumen can be displayed using three-dimensional (3D) printed models or augmented reality (AR)-based ones. It would enable hands-on evaluation or real-life device fitting, for 3D printing and precise quantification of geometry or multi-phase assessment, for AR models.

[3] `https://www.heartflow.com/heartflow-ffrct-analysis/`

For the segmentation, works such as the one by Huang et al. [35], the one by Cui et al. [40], or by Kong et al. [41] present methodologies based on different techniques, such as 3D U-net networks, in the first case; growing algorithms, in the second one; or a tree-structured convolutional gated recurrent unit, in the latter. Their Dice coefficient ranged between 0.7 and 0.87 in both cases. In 2020, ASOCA challenge was presented. The objective of participating teams was to develop a fully-automated segmentation of the coronary arterial lumen. Best performing teams achieved Dice coefficient values of 0.867 using nnU-Net as backbone [18].

For atherosclerotic plaque, some methods allow automatic classification of images according to the presence of calcified and non-calcified plaque, and in some cases, mixed plaque. Among these methods, some of which can be found in the review by Hampe et al. [42]. In other works, Zhao et al. [43], and Zuluaga et al. [44] presented different support vector machine algorithms aimed to classify images depending on the plaque composition. Focusing on the calcified plaque, although calcium scoring is not a complicated task *per se*, especially for a trained clinician, can be tedious and time-consuming when an extensive data set of images is under revision. For it, several methods have been already proposed, such as the ones by Cano-Espinosa et al. [45] and de Vos et al. [46], consisting of convolutional neural networks and allowing the calcium scoring in less than 1 second. Finally, regarding the adipose tissue surrounding the vessels, known as perivascular adipose tissue or PVAT, there are not segmentation methods in the literature available, although its presence is a proven risk factor of coronary disease [47].

Thus, coronary vessel segmentation is a point in the cardiac imaging pipeline where the introduction of AI can be a significant step forward. In fact, AI-based algorithms have already been introduced in clinical practice, where there already exists software used on a daily basis that allows the automatic identification of the main cardiac structures, such as the atrial and ventricular cavities, for example.

Clinical translation of AI

In cardiac image processing, every visualization software encompasses these AI-based tools and other classic image processing ones to allow automatic pre-, post-processing, and automated feature extraction; such as Vitrea, from Canon[4] or Intellispace Portal, from Philips[5]. They enable visualization, easy segmentation, and analysis of coronary arteries. Apart from that, there are external engines that aim to accelerate the development and deployment of such AI models by providing tools to create labeled datasets for the training of robust AI models and software to implement them, such as Graphics Processing Unit (GPU)-based NVIDIA Clara Medical Imaging[6]. In this regard, hardware acceleration, such as GPU-based acceleration, makes it possible to perform calculations involving massive amounts of data in a fraction of the time it would have taken on a classical CPU.

Time efficiency and the ability to extract computational markers of disease are some of the most significant assets of the platforms mentioned above, allowing physicians to concentrate on diagnosis rather than tedious manual segmentation. This encourages physicians to shift their focus to more complex tasks that are more difficult to adapt to AI-based workflows.

1.4 Objectives

Data processing and segmentation in clinical environments are currently implemented using many different software, which can be hard to combine or integrate into some workflows. Furthermore, the main limitation of most of them is their black box structure, not allowing to know the underlying segmentation algorithms and making impossible the adaptation to a concrete task since they are intended only for a limited set of actions. The implications of this problem involve the impossibility of adding innovative features or algorithms if the need arises, depending only on the willingness of vendors driven by economic interests.

[4] https://www.vitalimages.com/vitrea/

[5] https://www.usa.philips.com/healthcare/product/HC881103/intellispace-portal-11

[6] https://developer.nvidia.com/clara-medical-imaging

One of the most important bottlenecks in cardiac imaging workflow is the segmentation, especially its correction, adaptation to the subsequent steps in the pipeline, and posterior analysis. In this regard, the development of its algorithms and workflows would represent a significant advance for hospitals. It would enable them to be adapted to the specific needs of the professionals and the characteristics of each center and its patients while substantially reducing the time that the physician spends in performing repetitive manual tasks easy to learn for machines.

On the whole, this work aimed to develop and incorporate an AI-based pipeline that enables advanced DL-based algorithms to process multi-modality image data in cardiology in a clinical environment. For demonstration purposes, the segmentation of coronary vessels from CCT images was taken as a case for use. In addition, a proof-of-concept was carried out to evaluate its performance, and different deployment options were analyzed.

This project has been carried out in collaboration with the Hospital de Sant Pau. As a result, I have been able to integrate into the cardiac imaging unit team, where I have experienced the day-to-day from a clinical point of view. This has helped me learn first-hand about the existing problems and receive constant feedback from professionals in this field.

Chapter 2

Methods

This section aims to present the methodological workflow developed in this thesis, a pipeline that facilitates the processing of multimodal data in cardiology based on advanced AI algorithms. In addition, for demonstrative purposes, coronary artery segmentation is studied as a use case. Figure 7 shows the general pipeline developed in this work, including the input data, the different modules encompassed, the relations among them, and their outputs.

2.1 General workflow

The objective of the workflow is to process multi-modality imaging data. Therefore, the CCT images that were inputted into the pipeline were patient images stored in the PACS (Picture Archiving and Communication System) of the hospital. This computer system allows the storage, management, and retrieval of image files obtained from the source imaging equipment. It also performs the conversion from raw format to DICOM to allow their visualization on a hospital workstation by means of multiple medical image visors (3DSlicer[1], Horos[2], RadiAnt viewer[3], etc.). Thus, these DICOM images are the ones to be part of the input, but also, it can include other types of data, such as familiar or demographic data or segmentations.

[1] https://www.slicer.org/
[2] https://horosproject.org/
[3] https://www.radiantviewer.com/

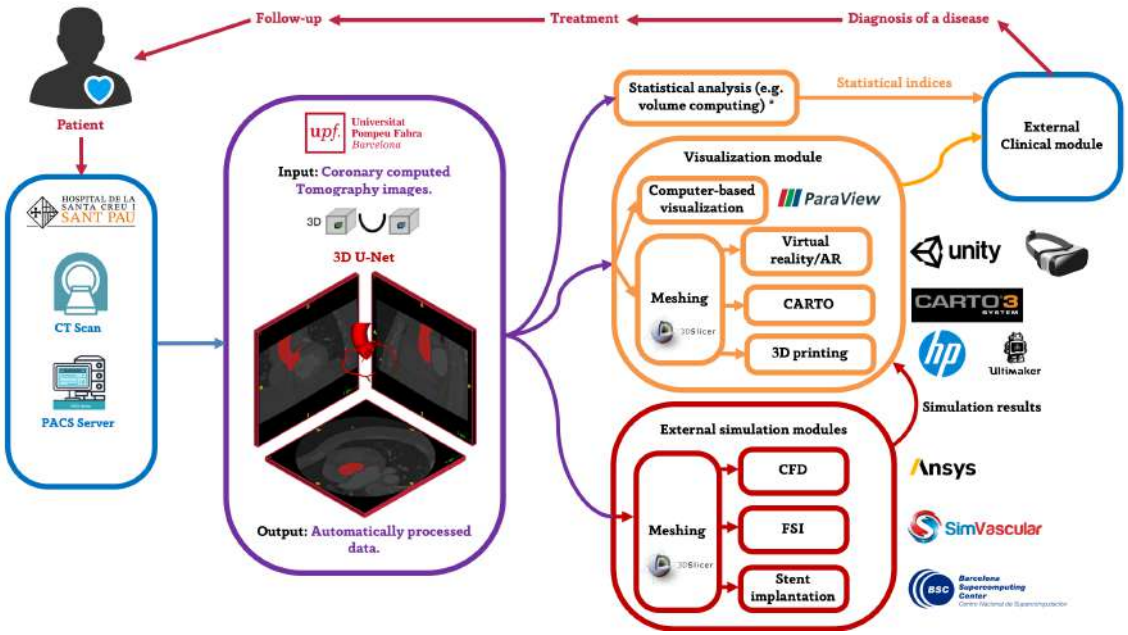


Figure 7: Summary of the thesis workflow adapted to the coronary artery segmentation task. Counterclockwise, starting from the upper left corner: patient undergoes cardiac CT studies, whose images are stored in the PACS (blue box); they are then subjected to segmentation algorithms to achieve segmentation of the structure of interest (purple box); the segmentation can then be used to compute statistical analyses on them or visualize them (yellow boxes) or to compute simulations on meshes created from them (red box); then, the output of each module is available to clinicians for diagnosing the patient; finally, the patient undergoes treatment, and some follow-up images can be later obtained. CT: Computed Tomography, PACS: Picture Archiving and Communication System, AR: Augmented Reality, CFD: Computational Fluid Dynamics, FSI: Fluid-Solid Interaction.

2.2 Description of workflow modules

2.2.1 Computational module

Once the images have been entered the workflow, they have to be processed, i.e., the critical information has to be extracted automatically. While it is true that medical imaging provides much information, most of it is not used. This is due to its lack of relevance to the clinical problem being studied, because it exceeds the processing capacity of the cardiac imaging specialist, or because of the limited time he/she can devote to each patient. Thus, to focus on the structure of interest and quantification purposes, the first step should be to segment the image. This process

is carried out in the Computational Module, together with image pre-processing, to adapt the data to the segmentation architectures both in image format and in the disposition of the images within the necessary folders. In addition, other types of data regularization, such as image normalization or standardization, can also be applied to the input images.

In parallel, it is also essential to apply some quality control, especially for some image modalities, such as magnetic resonance imaging. In those cases, automatic motion artifact detectors, such as the one proposed by Oksuz et al. [48], could be applied as an intermediate step. On other occasions, such as in CCT, a visual inspection by a qualified professional might be sufficient to assess a minimum quality and discard possible artifacts.

Afterwards, images can be fed to the corresponding segmentation algorithms; however, the latter must first be trained. In order to carry out the training of the algorithms, to later conduct the segmentation automatically, it is necessary to have a ground truth, i.e., the actual segmentations obtained from a certain set of images. These sets of images are usually obtained from data provided in challenges[4], such as the Medical Segmentation Decathlon[5]. If no dataset is available, one has to be created, using the available images and annotating them manually.

The training of the segmentation algorithm is the next step, which consists of learning the parameters that allow its best performance. However, this training is not a one-time process, as it must be iteratively trained with new data on a routine basis, allowing for a finer performance. Regarding the selection of the segmentation algorithm architecture, given that, at present, those that achieve state-of-the-art performance are those based on U-Net backbones, the nnU-Net could be the optimal selection due to its ability to self-configurate. After each training, the obtained model is stored, so that the following segmentations can be performed with it.

At the same time, these segmentations, which are in the format provided by the nnU-Net, are subsequently converted to adapt them to the input of the subsequent

[4] `https://grand-challenge.org/challenges/`
[5] `http://medicaldecathlon.com/`

modules. Since the specified output format of the images is 3D Neuroimaging Informatics Technology Initiative (NIfTI)[6], there is a need to convert them back into stereolithography (STL) format to be used in 3D viewing, AR-, and printing-based applications, computer-assisted simulations, and other different implementations, some of which are explored in the following sections.

2.2.2 Visualization and post-processing module

Once the information on the location and boundaries of the structures of interest has been extracted automatically, one possible next step is their visualization. It allows checking whether the reconstruction has been performed correctly and make the necessary corrections to ensure full accuracy in the subsequent steps.

There exist numerous solutions intending to enable the visualization of medical images. Some of the most used open-source ones, suah as Slicer or ITK-SNAP[7] allow not only to visualize the obtained segments superimposed to the original image but also, and more importantly, to conduct the necessary corrections. Other software are MITK[8] or VMTK [9], more adapted to blood vessels image processing. Some of them additionally allow being combined with AI-assisted annotation tools, such as NVIDIA Clara[10], providing additional segmentation and correction tools.

Afterwards, these segmentations are then used to create 3D meshes, which can serve as a domain to run fluid simulations or to be displayed in a computer visualization. This visualization is fundamental since it allows detecting problems or inconsistencies in the meshing. This way, it would be ideal for repairing and correcting the mesh while visualizing it; by correctly closing the surface, smoothing it, deleting intersecting faces, or changing the size of the elements. For this, some powerful free-access software can be used: Autodesk Meshmixer[11] or Meshlab[12].

[6] https://nifti.nimh.nih.gov/
[7] http://www.itksnap.org/pmwiki/pmwiki.php
[8] https://www.mitk.org/wiki/The_Medical_Imaging_Interaction_Toolkit_(MITK)
[9] http://www.vmtk.org/
[10] https://docs.nvidia.com/clara/tlt-mi/aiaa/index.html
[11] https://www.meshmixer.com/
[12] https://www.meshlab.net/

It is essential to emphasize the importance of this step, the mesh processing, since most of the work implemented in the subsequent modules depend on its quality. It is also important to note that mesh processing can be done in entirely different ways depending on the intended use of the mesh. For example, for CFD-based blood flow simulations in a blood vessel wall or heart cavity, a 3D volumetric mesh should be firstly constructed, being smooth enough to ensure convergence. In addition, some structures could be added to the mesh to facilitate the application of boundary conditions. On the other hand, for surgery planning or display applications, the processing should adapt the mesh to the actual contour of the corresponding structures, even if it would be impossible to compute simulations on it.

Augmented reality, virtual reality and 3D printing

In recent years, the field of computer-aided visualization has increased significantly, implying that, for accurate viewing of 3D structures, simple computer vision software may not be precise enough. It is essential to have an immersive experience, know how the 3D object looks like in shape and size, and, in the case of surgery planning, know its location and how to access it. In this sense, the development of AR and Virtual Reality (VR) tools; such as Unity[13] or AR glasses[14], allow for preoperative planning and checking during the intervention if everything is going as planned.

In addition, to provide a more sensory point of view, these tools could be combined with patient-specific 3D printed models, which are easier to work with and allow a clearer idea of the shape and size of the actual structure or embedded lesion. With this objective, currently, many hospitals are introducing 3D laboratories into clinical practice, such as Parc Taulí[15], which highlights the importance of this knowledge for doctors. In addition, another use for 3D printed models could be the validation of computer simulations, by using patient-specific phantoms [49]. Therefore, the combination of these different tools depending on the problem to be treated should be possible, allowing to select those that could be more useful.

[13] https://unity.com/
[14] https://news.microsoft.com/de-ch/2020/12/11/hololens-in-surgery/
[15] https://www.tauli.cat/en/institut/plataformes-i-serveis/laboratori-3d/

2.2.3 Statistical analysis module

After the automatic segmentation of structures and their corresponding correction and post-processing, the next step is to use this information to extract useful quantitative information to perform diagnosis. Some of the routine tasks that physicians perform daily consist of determining the shape and position of certain structures and perform measurements of diameters or volumes. Therefore, not only can the first part be automated employing automatic segmentation algorithms, but there are also ways to perform these measurements and calculations without manual intervention.

In this sense, it would be much straightforward to characterize diseases whose diagnosis is based on the presence or not of some determined computational markers, including those measurements of lengths or volumes conducted on body structures. Thereby, the statistical analysis module would output numerical data to be used by the physician when diagnosing or assessing the condition of a patient.

2.2.4 External modules

After segmentation of the structures and extracting the functional information from various imaging modalities, it can be used for many different tasks. One of the most important is to use the segmented anatomical data, which can be combined with the available data to calculate finite element method (FEM), CFD, or fluid-solid interaction (FSI) simulations. In this way, parameters that are impossible or risky to assess by any imaging modality can be more easily and safely obtained in a patient-specific manner. In addition, procedures such as planning invasive surgeries or the implantation of devices would also benefit from the information acquired.

2.2.5 Clinical module

Once the images have been segmented, and the crucial information has been extracted, these data were exported to the clinical module, where they are stored and available for use by the physician when making the diagnosis of the patient. This module not only stores automatically obtained data, but it also includes man-

ually introduced information, such as the weight or the height of the patient, or demographic data. In other words, this module contains all the information that is necessary for diagnosing the patient. Moreover, in the medium-long term, this information could train AI-based algorithms to perform the diagnosis automatically.

2.3 Use case: Coronary artery segmentation

This section aims to describe the workflow followed for the use case studied and implemented in this thesis: the automatic segmentation of coronary artery from CCT images. It has been implemented in Hospital de la Santa Creu i Sant Pau[16] (Barcelona, Spain). It is a hospital center in which numerous projects related to CT in Cardiology have been carried out in recent years, being one of the pioneering centers in Spain in performing CCT studies. As a result, the PACS of the hospital gathers a significant number of those studies: more than 25,000. Therefore, their use would allow the creation of a large database of segmentations.

Regarding the coronary arterial lumen segmentation, several considerations must to be taken into account. First, they are tiny vessels and therefore appear very small in CCT images, as Figure 8 displays. This may be a problem, as relatively small differences in annotations by different practitioners could significantly impact the segmentation models, which is not the case in larger structures. Additionally, other problems such as artifacts related to cardiac or respiratory motion, or the presence of both calcified and non-calcified plaque can impede the correct segmentation of the lumen.

2.3.1 Coronary dataset creation

To train the coronary segmentation algorithm, it is necessary to input a dataset consisting of both the images and the corresponding binary masks of the coronary arterial lumen. Since there is no freely accessible data set (and access to the ASOCA challenge dataset was not possible since organisers did not answer our requests), it was necessary to generate one from scratch. The objective of this first dataset was to train a simple model to check the strong and weak points of the segmentation algorithm and, based on the results, iteratively improve the model. The procedures followed for the creation of the first dataset are outlined in this section.

[16] `http://www.santpau.cat/en/web/public`

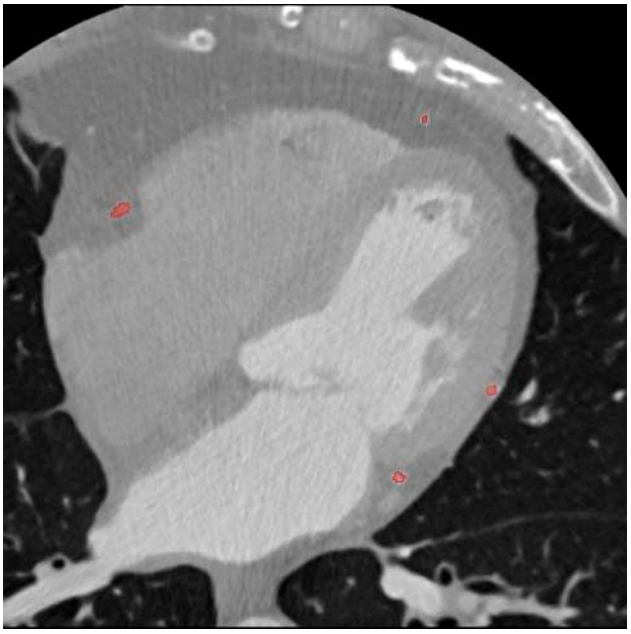


Figure 8: Transverse cut of a cardiac computed tomography volume highlighting the coronary arterial lumen in red.

The first step was the selection of the images. Forty images were obtained, each from a different patient, using contrasted CCT at around 75% of the R-R interval, corresponding to mid-end diastole. Due to the differences in the anatomy of the patients, each image had a different number of slices and spacing among them while preserving the slice size (being it 512x512 voxels). The images were obtained from 2006 to 2015 from two different vendors: Toshiba Medical Systems[17] (currently belonging to Canon Medical Systems Corporation) and Philips Healthcare[18]. Table 1 collects the information such as the CT model and the institution, the number of training and testing data obtained from each of them.

The selection of two different vendors has been conducted based on the fact that the software at *Hospital de Sant Pau* is aimed at the analysis of patients from the same hospital and *Clínica Creu Blanca*[19]. Since both centers have different CT scanner manufacturers, CCT images from both centers have been selected without giving paramount importance to any of them. This decision has been taken considering

[17] `https://us.medical.canon/products/computed-tomography/`

[18] `https://www.usa.philips.com/healthcare/solutions/computed-tomography/ct-scanners`

[19] `https://www.clinicacreublanca.es/en`

that, in the case of only training the algorithms using images from only one of the centers, the generalization would have been poor, according to recent studies [50].

Vendor	Toshiba-Canon	Philips
Institution	Clinica Creu Blanca	Hospital de Sant Pau
CT scanner model	TOSHIBA Aquilion ONE	Philips iCT 256
Acquisition date interval	17/January/2006-11/April/2013	07/May/2010-19/August/2015
Total data	11	29
Training data	8	21
Testing data	3	8

Table 1: Information about the multi-vendor CCT dataset. CT: Computed Tomography. CCT: Cardiac Computed Tomography.

These images were gathered in the PACS of the hospital in DICOM format, from where they were uploaded to a visualization platform, Philips Intellispace Portal 9, after being inspected by a cardiologist to discard major artifacts and ensure a minimal quality. This platform includes an automatic segmentation tool, CT Comprehensive Cardiac Analysis[20]. It is an engine with different functions to assess the state of the heart, including the automatical detection and segmentation of cardiac structures, such as coronary arteries, and some correction manual tools to enhance segmentation. Using this software, the segmentation was first created and then exported in STL format, as Figure 9 displays.

Once the DICOM images and the STL segmentations were available, the next step was to convert the format to one that could be input into the corresponding segmentation method. Since the selected method is the nnU-Net, the input format must be similar to the one used in the Medical Segmentation Decathlon, being all the input images compressed 3D NIfTI files (.nii.gz). For so, it was necessary to carry out the corresponding format conversions, first, from DICOM to NIfTI for the images and then from STL to NIfTI for the segmentations.

[20] https://www.philips.co.uk/healthcare/product/HCAPP006/-ct-comprehensive-cardiac-analysis-cca-

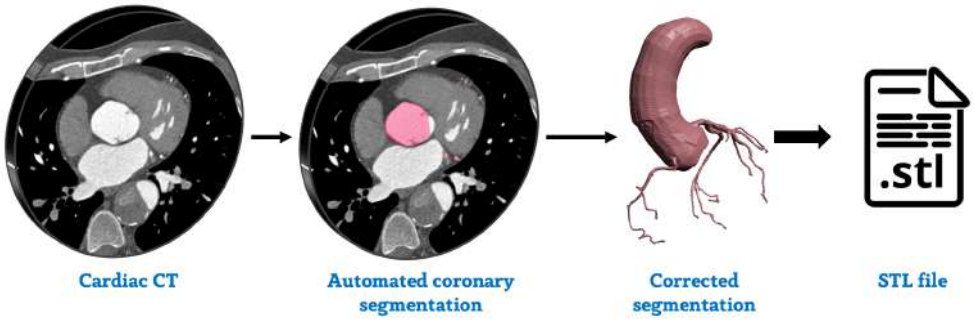


Figure 9: Workflow for the dataset creation. The first three steps were conducted within the CT Comprehensive Cardiac Analysis software, in order to create the final STL segmentation, being it exported to the device. CT: computed tomography.

The first conversion was achieved with the help of the Python library dicom2nifti 2.2.12[21] using a Python script, while the second one was done by integrating the Slicer medical image processing and visualization system into the Jupyter[22] environment by using the Slicer module named SlicerJupyter[23] [51], [52]. The implementation was carried out in a script that allowed first to load the segmentation to Slicer and then transform it into a label map with the exact dimensions as the entire image volume. Finally, the label map was exported to a compressed NIfTI file to generate the dataset composed of image+label map pairs. It is important to emphasize that the NIfTI files created contain only the intensity information of each voxel and metadata related to image quality and size. Thus, converting the images into NIfTI format also allows for complete anonymization of the data.

For the preparation of the data to be trained, since they had to be arranged in a specific fashion so that the nnU-Net would know how to load and interpret them, the steps indicated by Isensee et al. [17] were followed. These steps included placing the data in the appropriate folders and creating a JavaScript Object Notation (JSON) file. Figure 10 shows the final disposition of the data within the folders. After this step, the final folders containing the correctly stored images were sent to the HPC (High-Performance Computing) cluster at the UPF to train the network using a GPU. There, the data were pre-processed, and then the training was performed.

[21] https://github.com/icometrix/dicom2nifti
[22] https://jupyter.org/
[23] https://github.com/Slicer/SlicerJupyter

```
nnUNet_raw_data_base/nnUNet_raw_data/Task001_CoronaryArteries/
├── dataset.json
├── imagesTr
│   ├── CoronaryArteries_004_0000.nii.gz
│   ├── CoronaryArteries_005_0000.nii.gz
│   ├── ...
├── imagesTs
│   ├── CoronaryArteries_001_0000.nii.gz
│   ├── CoronaryArteries_002_0000.nii.gz
│   ├── ...
└── labelsTr
    ├── CoronaryArteries_004.nii.gz
    ├── CoronaryArteries_005.nii.gz
    └── ...
```

Figure 10: Disposition of the data within the folders to allow its compatibility within the nnU-Net framework. imagesTr stores the cardiac computed tomography images of the training set, imagesTs contains the images of the test set and labelsTr stores the ground truth segmentations for the images in the training set.

Data pre-processing, training and testing:

Firstly, the raw data was automatically pre-processed based on the dataset-specific properties extracted by the nnU-Net framework, which is a low-dimensional representation of each dataset's attributes. Then, these attributes are summarized in the so-called *dataset fingerprint*, which includes, among other data, image modality, image size, and voxel spacing. The determination of this dataset fingerprint is the first step in subsequently inferring the pipeline fingerprint, which consists of the critical design choices of the segmentation algorithm, being divided into the blueprint, inferred, and empirical parameters. First of all, the blueprint parameters control fundamental design options and hyper-parameter selection. Second, the inferred parameters are responsible for adapting the environment to a given dataset by pre-processing, encoding the network architecture, and determining the batch size and the patch size. Finally, the identification of the empirical parameters is made by cross-validation of the training cases. In this particular case, for the training of a network able to segment the coronary arteries from CCT images, a full-resolution 3D UNet was selected among the different network architectures available within the nnU-Net framework because of its efficiency and the best performing configuration, according to Isensee et al. [17].

2.3.2 Design of the segmentation model

nnU-Net with original size images

The first training was conducted for the first coronary dataset with this configuration, whose training set was composed of 29 3D images at the UPF HPC cluster. There, GPU-accelerated training was carried out by making use of the multiple available cores and GPUs[24]. Due to time constraints, the number of epochs was set to 500, instead of the default 1000 epochs, with an *epoch* being defined as the iteration over 250 training batches. After training the segmentation model, the inference was performed, and the resulting segmentations were extracted from the cluster for analysis.

Concerning the network architecture used, it was a six-stage U-Net, with two blocks per stage, each consisting of a convolution layer, an instance normalization [53] and a non-linearity layer, with the selected nonlinear function being the leaky ReLU (rectified linear unit). The down-sampling of the network used in the encoder was achieved by 2-strided convolutions, whereas, in the decoder, up-sampling was achieved using transposed convolutions. The output of all these transposed convolutions had the same shape as the skip connections originating at the encoder. The patch size for this network was 96x160x160 voxels, while the batch size was 2. Figure 11 shows a schematic representation of this network, indicating the kernel sizes and strides for each block.

The analysis performed for the automatically achieved segmentations consisted of a detailed and individualized description of both the problems encountered and the positive aspects to be highlighted in each case in which the inference was applied. It was carried out under the close advice of cardiac imaging professionals from the Hospital de Sant Pau. After this individual analysis, the most common problems were collected. In this way, a new problem-based analysis was performed. In it, the aspects discussed were the prevalence (the number of cases in which the segmentation problem was present), the probability of recurrence (which is the probability of the

[24] `https://guiesbibtic.upf.edu/recerca/hpc/system-overview`

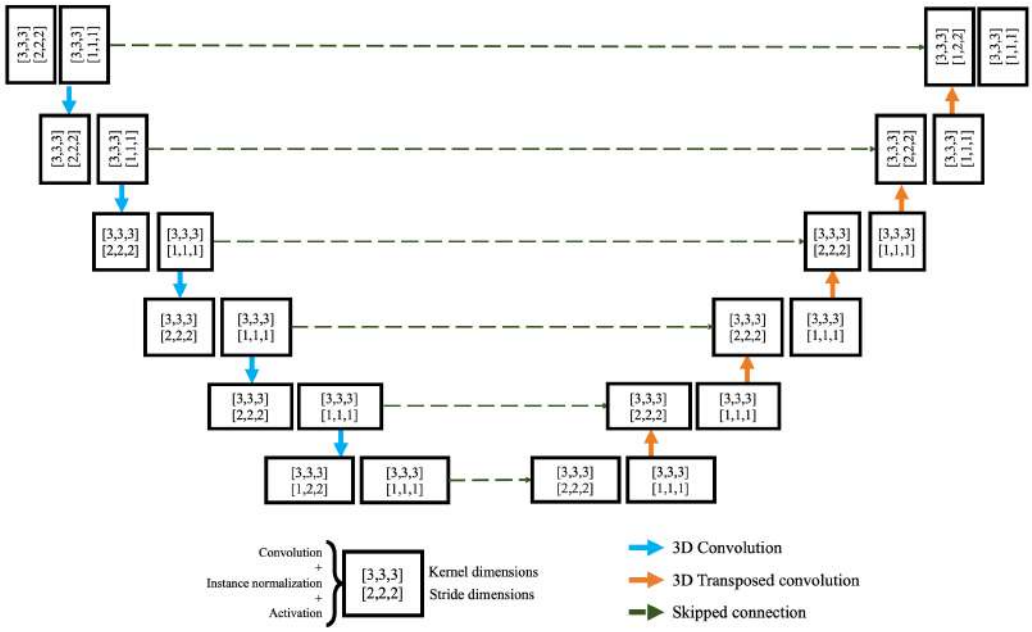


Figure 11: Full resolution 3D U-Net architecture trained on the nnU-Net framework and used for the coronary dataset. Each block contains a convolution, an instance normalization and a leaky ReLU activation function.

error occurring in general in an unseen case), and the severity of the problem (being the degree of severity of the possible consequences of a case presenting this problem). In addition to these elements to be considered, the analysis sought to explain the causes of each of these problems and to propose some potential solutions to them. The Dice coefficient (defined in Equation 2.1, where A is the ground truth and B the obtained segmentation) was also assessed.

$$\text{Dice coefficient} = \frac{2|A \cap B|}{|A| + |B|} \tag{2.1}$$

External vessels discarding

The second training was carried out after conducting the analysis with the objective of bypassing one of the most frequent problems. Most segmentation algorithms, especially those based on convolutional layers (such as the U-Net), are based at the lowest level on edge detection filters. Therefore, in this work, the structures to be detected were tubular ones with intensity values around 350 Hounsfield Units (HU),

depending on factors such as the concentration of the contrast agent or the cardiac index. Considering that the CCT images used in this work contain a volume larger than only the pericardial cavity, other vessels with similar characteristics could be erroneously segmented by the algorithm, e.g., mammary or pulmonary arteries (see Figure 12). Consequently, to avoid these problems, which would lead to inaccurate results, an experimental set-up was developed, explained below.

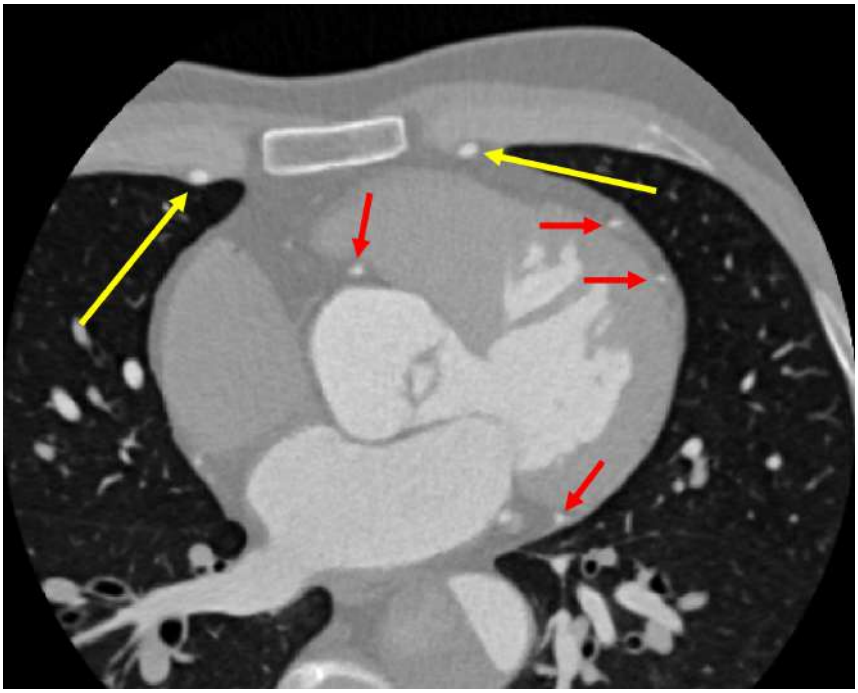


Figure 12: Example slice of a cardiac computed tomography volume displaying two types of arteries. Pointed out in red: some of the coronary arteries that should be segmented. Pointed out in yellow: mammary arteries that should not be segmented.

Since the coronary arteries are located inside the pericardial sac, contrary to the vessels that the algorithm should not segment, a solution to improve the quality of the segmentation could be to exclude from the final segmentation those vessels external to the pericardium. To apply it, a nnU-Net, different from that used to achieve the segmentation of the coronary vessels, was trained at the UPF HPC cluster. The architecture of the network, the batch size, and the patch size were the same as the U-Net used to train the algorithm for the coronary artery segmentation (see Figure 11). Its objective was to perform automatic segmentation of the pericardium to later apply a Boolean intersection (logic gate AND) between this segmentation and that of the vessels to keep only the vessels inside the pericardial sac.

Considering the role of pericardial volume in the final model, some considerations for the dataset on which the training were performed can be extracted. The most important one is that all the volume outside the pericardial sac was removed from the final segmentation. This way, it is crucial that the pericardial segmentation does not leave out any coronary vessel, i.e., the sensibility of the segmentation model should be as high as possible, not allowing false negatives (FN). It is essential in the locations where the coronary artery is located right next to the pericardial sac, as shown in Figure 13.

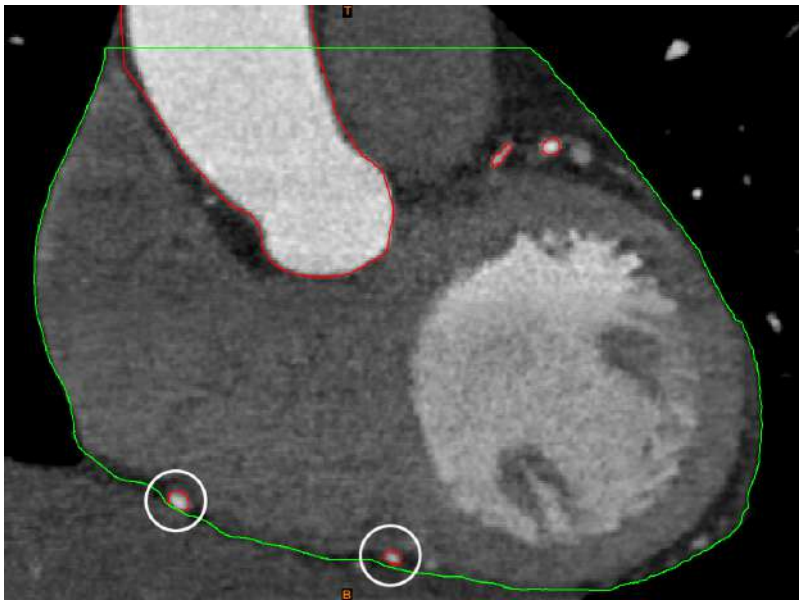


Figure 13: Coronal plane of a cardiac computed tomography. Red: contour of the coronary artery. Green: contour of the pericardial sac. White: sections where two coronary vessels are located contiguous to the pericardial contour.

Concerning false positives (FP), they are not as harmful as FN, as vessels close to the pericardium that could be considered coronary ones after the intersection between both segmentations are easier to discard upon subsequent post-processing. Thus, when creating the dataset, it was more favorable to slightly expand the contour of the pericardium, avoiding potentially damaging FN.

With these instructions in mind, the dataset was created. The first pericardial segmentation model was trained and tested with the same images as the first coronary arteries segmentation model, so it was easier to perform the Boolean intersection between both segmentations and thus see the final result. In order to create the

ground truth segmentations, the software Materialise Mimics Innovation Suite 23[25] was the one used. For each image, a mask was created in which voxels whose HU were greater than -300 were selected with the objective of identifying the boundary between the pericardium and the pleural cavity (the latter with HU values lower than -300). With this in mind, multi-slice editing was used to create the segmentation of the pericardium. This segmentation consisted of interpolating voxels with HU>-300 between some manually segmented slices and thus creating a volume (Figure 14 show an example for multi-slice editing). Finally, each segmentation was exported in STL format.

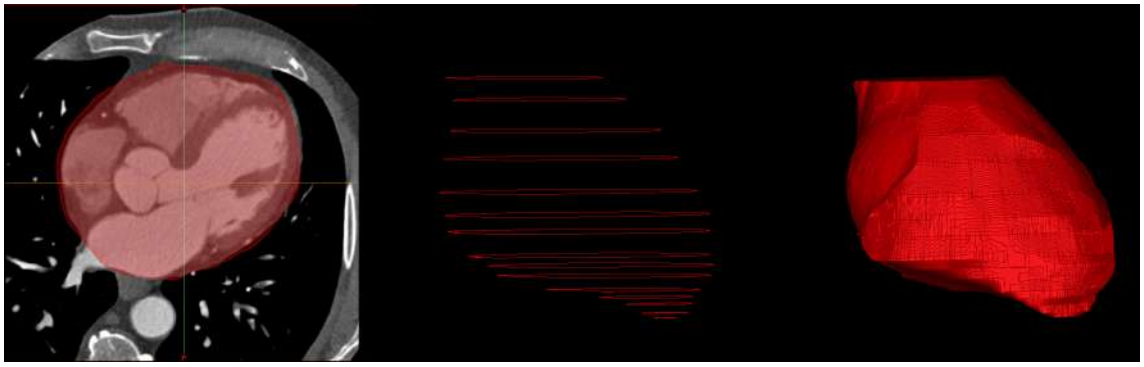


Figure 14: Manually annotated slices, used in multi-slice editing. Left: transverse cut of a cardiac computed tomography volume. Center: 3D view of the contour of the manually annotated slices. Right: interpolation of these slices.

Afterwards, the STL files were converted to NIfTI by using the same procedure as in the case of the coronary arteries, and the dataset was constructed following the same steps, as displayed in Figure 15. Subsequently, after assembling the dataset, it was sent to the UPF HPC cluster to use the GPUs available there to train the DL-based algorithm. There, the data was automatically pre-processed, and then, the network was trained. The selected pipeline within the nnU-Net framework was a full-resolution 3D U-Net, and the selected number of epochs was 500, similarly to the previous case. After the training, inference for the test set was run, and data was extracted for analysis.

[25] `https://www.materialise.com/en/medical/mimics-innovation-suite/23`

```
nnUNet_raw_data_base/nnUNet_raw_data/
├── Task001_CoronaryArteries
│   ├── dataset.json
│   ├── imagesTr
│   ├── (imagesTs)
│   └── labelsTr
├── Task002_Pericardium
│   ├── dataset.json
│   ├── imagesTr
│   ├── (imagesTs)
│   └── labelsTr
├── ...
```

Figure 15: Example folder structure for the two tasks to make data compatible with nnU-Net framework.

CCT volumes down-sampling

As previously discussed, the CCT volumes used to achieve the automatic segmentation were composed of several slices with a size of 512x512 voxels each. Thus, each slice has more than 250,000 voxels, with the total number of voxels in some images exceeding 150 million, leading to very high computational costs and long training times. Therefore, an experiment that incorporated a dataset's size reduction was proposed and analyzed.

Precisely, this methodology consisted of down-sampling each slice to the same size, enabling the volume to be more manageable. However, the more the size of an image is reduced, the fewer details it has. Therefore, this analysis was carried out to test: firstly, whether the down-sampling procedure achieves a significant reduction in computational costs and; secondly, if so, what are the optimal slice dimensions that allow these costs to be reduced while retaining good image quality and segmentation capability of the coronary arteries.

The selected slice sizes to be analyzed were 256x256 (a 2x2 down-sampling), 128x128 (a 4x4 down-sampling), and 64x64 (an 8x8 down-sampling), and the resizing was achieved after the conversion from DICOM to NIfTI with the *transform* module of the *sci-kit image* toolbox for Python[26] [54]. After the down-sampling, the procedure followed in creating the different datasets was similar to those followed previously in the pericardium and coronary datasets. A full-resolution 3D U-Net was then trained

[26]https://scikit-image.org/docs/dev/api/skimage.transform.html

for 500 epochs for each of these data sets at the UPF HPC cluster. This network had the same architecture as the U-Net used in the training of the coronary artery and the pericardium segmentation algorithms, which is displayed in Figure 11. However, in this case, the patch size was 160x128x112. The inference was subsequently run for each of them, and the results obtained were analyzed as previously detailed.

2.3.3 Final segmentation model

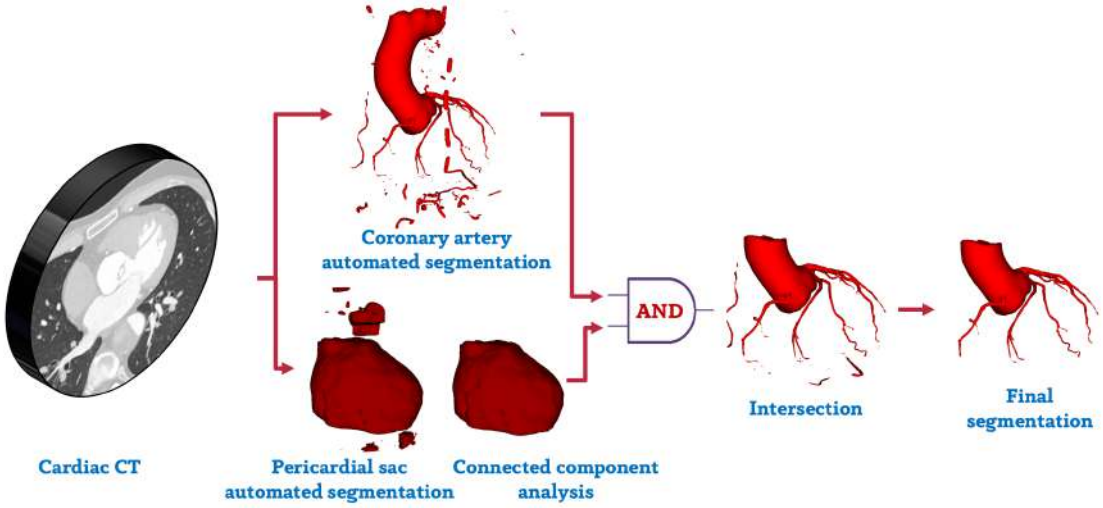


Figure 16: General workflow for the segmentation model. Two different neural networks were trained from the cardiac computed tomography volume: one for the coronary artery segmentation and the other one for the pericardial sac segmentation. The pericardial sac is then post-processed, employing the selection of the largest connected component, and the coronary segmentation is intersected with the result. Finally, a connected component analysis is carried out in the volume, and the largest component is selected, resulting in the final segmentation. CT: Computed tomography.

Eventually, the pipeline of the final segmentation model was elaborated based on the feedback provided by the inference using the previously trained experimental models. It consisted of the series of steps summarized in Figure 16. First, the images were converted from STL to NIfTI to allow their use by the nnU-Net framework. Then, each image was duplicated; one of them should be used to segment the coronary artery, and the other one was introduced into the pericardium segmentation model. Both models were analogous to those presented previously, as they were based on the full-resolution 3D U-Net provided in the nnU-Net framework and used original

size images. Training, in this case, was completed using 40 images in the training set for each network. Once both segmentations were achieved, the pericardium mask was cleaned up by preserving only the largest connected component. Afterwards, an intersection between the volume of the coronary and that of the pericardium was conducted. Finally, post-processing was applied, consisting of discarding the non-largest connected components.

2.3.4 Deployment of the model

For the deployment of the AI-based segmentation model, there are a few options. This section aims to describe these options, by which the trained model can be used from a hospital center.

Firstly, the hospital could have an AI unit with the necessary resources to carry out both algorithm training and testing operations and infer new data from these. This unit should have a sufficient number of CPUs and GPUs to train those state-of-the-art algorithms, given that they normally require high computational power. For the deployment per se, an affordable option could be the deployment of networks that can be run in CPU-based computers using executable builders, such as PyInstaller[27].

Another option could be the alliance of several hospitals with common characteristics (such as geographic location or specialization in a medical specialty) to develop an AI center that uses data from all these centers and can benefit. In this way, the cost per center would be lower, and the total computational power could be increased. This center could be an existing one, such as a university or a research center, that could share or rent some of its resources for clinical purposes.

Finally, another type of implementation would consist of leveraging those platforms that offer cloud computing services. In this way, it would not be necessary to purchase equipment based on multiple processing units but would use these platforms, paying according to the amount of computation performed. An example of a cloud-based platform would be Amazon Web Services (AWS)[28].

[27] `https://www.pyinstaller.org/`
[28] `https://aws.amazon.com/`

2.3.5 Proof-of-concept workflow

As a proof of concept, the workflow was evaluated for an example, which is summarized in this section and is fully covered on Appendix B. This workflow included all the steps presented theoretically, such as automatic segmentation and subsequent mesh generation, fluid simulation and different examples for the visualization of the results.

Segmentation, mesh generation and post-processing

The first step was to select a DICOM study that was not used for training the segmentation algorithm. Then, it was converted to NIfTI format and introduced into the nnU-Net framework to infer both the coronary arteries and the pericardium segmentations. Then, the final coronary artery volume was obtained, and it was exported in STL format. Subsequently, it was introduced to Slicer, where post-processing, including certain geometric modifications necessary to compute a fluid simulation, was performed. The post-processing was carried out following the advice of an expert cardiologist.

CFD simulation

For the simulation, it was necessary to know the boundary conditions that would be applied to it, so that the model could exhibit patient-specific behavior. In this way, both CCT images and Pulsed-Wave Doppler echocardiogram ones were used to compute the cardiac output. Knowing these data, the mesh was introduced into the simulation software, which was Simvascular[29], where both a surface and a volumetric mesh -the latter with 825,000 tetrahedral elements- were created. Then, the boundary conditions for each of the inlets and the outlets were specified, stating the waveform of the aortic inlet, a Windkessel RCR (resistance, capacitor, resistance) for the aortic outlet, a modified lumped parameter model for the coronary outlets and a no-slip and no-penetration boundary condition on the vessel walls. Furthermore, the parameters of the simulation were also specified and, with that information,

[29] `https://simvascular.github.io/index.html`

the simulation was run. The haemodynamic descriptors exported after running the simulation were the pressure, the velocity and the WSS.

Visualization of the results

The following step was to achieve an easy and appealing visualization of the variables exported. In this way, both the WSS, by means of coloring the wall depending on the WSS and the velocity, by means of the simulated blood flow lines, were displayed on computer software. In addition, other types of visualization, such as those based on 3D printing and AR, were analyzed. In order to create print-ready models that could also input into AR-specific software, the exported variables were post-processed by creating a colored volumetric model with patient-specific hemodynamic information based on CFD simulation.

Chapter 3

Results

3.1 Segmentation model

3.1.1 Dataset creation and first training

Coronary arteries dataset creation

Regarding coronary segmentation, the available commercial software from Philips that was used -CT Comprehensive Cardiac Analysis- presented some errors. Mainly, it could not correctly identify the most important coronary arteries in most cases, as it failed to recognize some of the vascular structures. For this reason, some post-processing was necessary in all the cases. The most common problem was the missing segmentation of a secondary coronary artery, which had to be added manually. Other less frequent but important problems were the misclassification of a vein as a coronary artery or the absence of a main coronary artery segmentation. Finally, sometimes, it was necessary to manually correct the borders of a coronary artery, which were not correctly detected by the software. As an illustration, Figure 17a displays an example of a segmentation obtained by CT Comprehensive Cardiac Analysis software from Philips. In this case, the anterior descending artery was not detected, needing to be manually identified before being incorporated by the software, and its tracing was subsequently corrected manually (Figure 17b).

(a) Segmentation before manually adding the anterior descending artery.

(b) Segmentation after manually adding the anterior descending artery.

Figure 17: Example of automatic segmentation before and after manually adding the anterior descending artery.

In total, the automatic segmentation of the heart (including the coronary artery) lasted around one minute, while the manual post-processing had a duration ranging from 15 to 20 minutes for a cardiac imaging expert. Figure 18 displays an example of segmentation included in the ground truth. Some of the segmentation characteristics of the CT Comprehensive Cardiac Analysis software, such as the poor segmentation of the aortic root (according to cardiac imaging experts), can be seen there.

Coronary arteries dataset training

The first experiment performed aiming to determine the final segmentation model consisted of training a full-resolution 3D-UNet using the nnU-Net framework and a training set size of 29 images during 500 epochs. It took seven days and 21 hours to train in the UPF HPC cluster.

After the training, the inference was the following step. The 11 NIfTI images from the test set were fed to the algorithm, and, in total, the process lasted 2 hours and 15 minutes in being computed. This means that each segmentation required approximately 12 minutes to be obtained. Figure 19 shows an example of an achieved segmentation (Figure 19a) together with the ground truth (Figure 19b). The main

differences between the ground truth and the automatic segmentation can be seen in two areas. Firstly, the automated segmentation of the ascending aorta and the aortic root was much more accurate and smooth than the ground truth, which was coarser. Secondly, the segmentation of the coronary vessels presented some stops in segments in which luminal segmentation should not be stopped, leading to unconnected segments in some vessels. In addition, other examples showing the segmentations before any post-processing are presented on Appendix A.

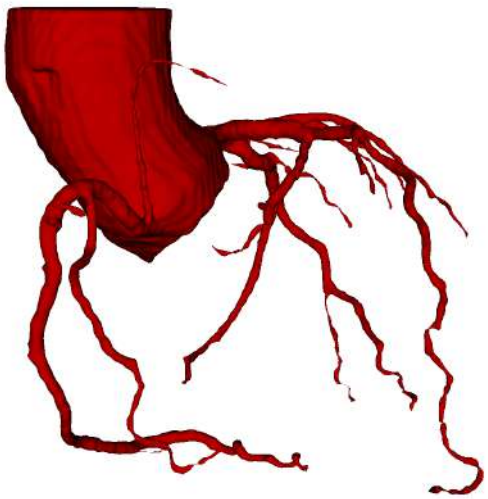
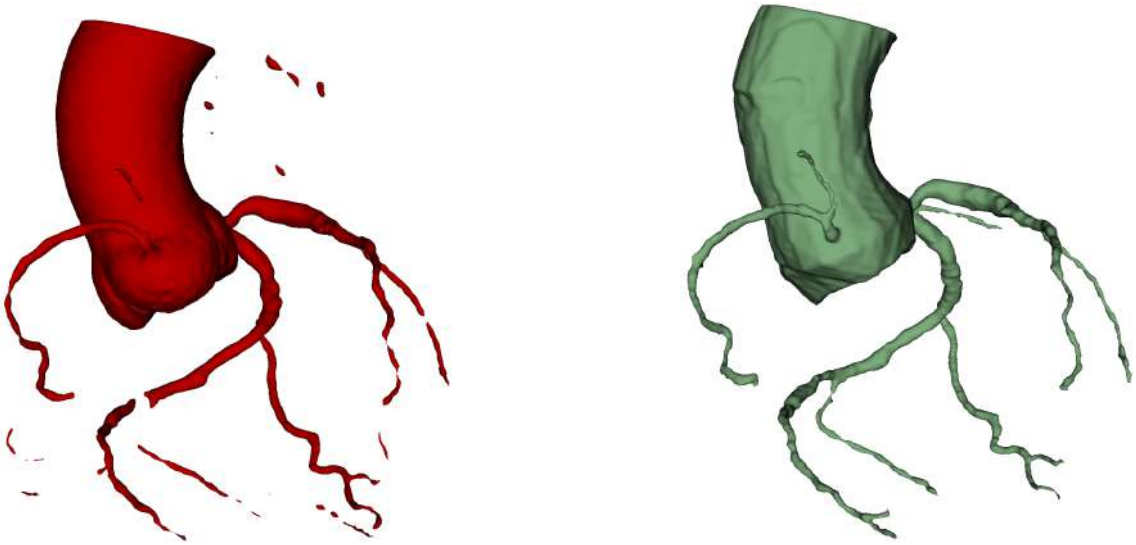


Figure 18: Example of ground truth included into the dataset.



(a) Automated segmentation. (b) Ground truth segmentation.

Figure 19: Automated segmentation compared to the ground truth.

Then, the binary masks were converted to 3D meshes using Slicer. When all the meshes were obtained, the performance of the framework was evaluated by comparing each of those meshes with the ground truth one. For it, the first analysis, i.e., the individual comparison, was computed, being the results presented on Appendix A. Afterward, the global analysis, which gathered the most common problems from the segmentation results, was conducted. Here, Table 2 show what these problems were and the number of cases in which they were. Figure 20 displays the number of cases in which these errors occur based on their importance. The complete analysis, including the probability of recurrence, possible causes, and potential solutions, is presented on Appendix A. The mean Dice coefficient for the 11 cases was 0.925±0.037.

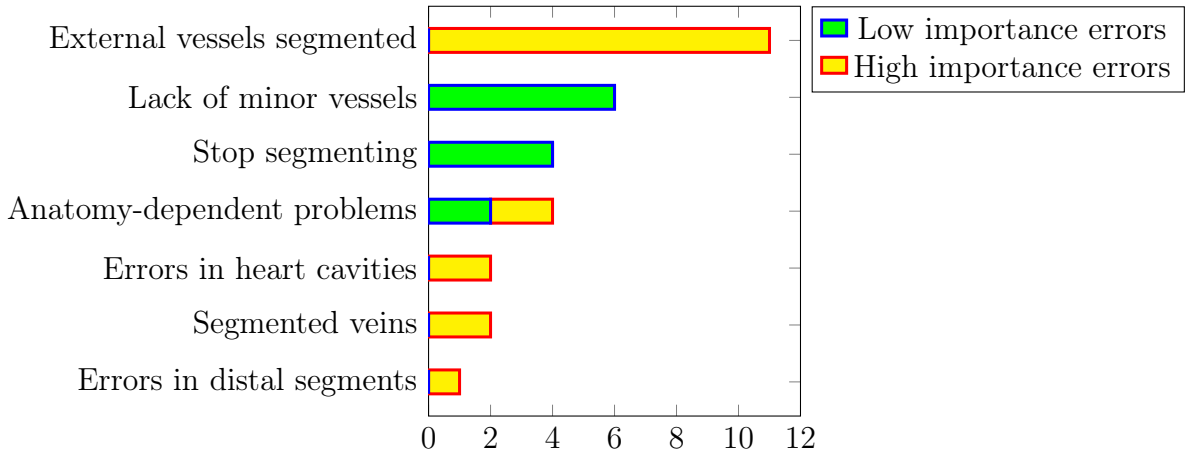


Figure 20: Classification of the errors of the automatic segmentation based on their importance.

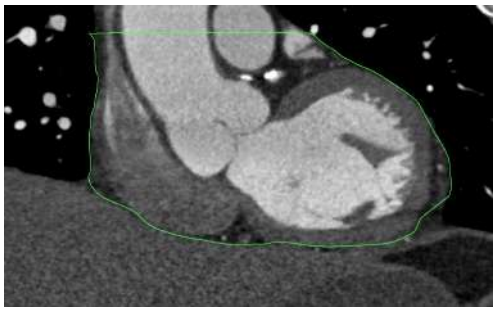
3.1.2 Design of the segmentation model

Pericardium dataset creation

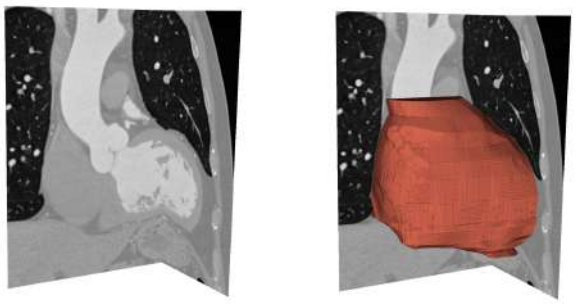
So as for the semi-automatic segmentation of the pericardial sac, Figure 21a shows an example of the final segmentation. In it, it can be seen how the interpolation managed to identify the contour of the pericardium despite manually selecting on average only about ten slices. In addition, Figure 21b displays a semi-automatically segmented pericardial sac superimposed to the coronal and the sagittal plane cuts of the corresponding CCT volume.

Detailed explanation of the error	No of cases (n=11)
External vessels to the pericardium were segmented.	11
Lack of segmentation of complete minor vessels: diagonal, acute marginal of the right ventricle, conal, posterior descending, marginal of the right coronary, marginal anterior descending.	6
It stopped segmenting in segments where it does not have to stop segmenting in the main vessels. It is not caused by a decrease in contrast (probably due to plaque) nor by a visible thickening of the arterial wall: in circumflex artery (x2) and in diagonal (x2).	4
Rare anatomy-dependent problems: myocardial bridging (x2), atrial diverticulum (x1), coronary artery abnormality (x1).	4
Segmentation of part of some heart cavities as coronary arteries: left atrial appendage (x1) and right ventricle (x1).	2
Segmented veins within the pericardium.	2
Errors when segmenting the most distal segment of certain small caliber vessels.	1

Table 2: Presentation of the most common segmentation problems encountered in the first coronary segmentation model.



(a) Green: Contour of a semi-automatically segmented pericardium in a sagittal view.



(b) Coronal and sagittal plane cuts of a cardiac computed tomography volume without (left) and with (right) the semi-automatically segmented pericardial sac superimposed to it.

Figure 21: Semi-automatically segmented pericardium. It can be seen how the interpolation perfectly fits the actual contour of the pericardial sac.

Pericardium dataset training

As seen in Table 2, the most commonly reported error that was present in each of the cases was the erroneous segmentation of extra-pericardial vessels. Therefore, the next step was to develop a pericardial segmentation model to identify its volume in

order to exclude these vessels. In this case, the training of the model was conducted in 8 days and 22 hours. In terms of inference, it took 4 hours and 28 minutes to calculate the 11 segmentations, which was about 24 minutes per case.

After obtaining the binary masks, each was converted into a mesh using Slicer, and the quality of the segmentation was assessed. In this case, the exact match between the ground truth and the segmentation was not the objective as it was much more important not to discard vessels that were actually within the pericardial sac volume. Because of this, the analysis was carried out by combining both the coronary artery and the pericardial segmentations.

The conducted analysis showed that in 2 cases out of 11, the model generated FN in the inferior part of the pericardium due to lack of segmentation, affecting the coronary arteries there. In 1 case out of 11, a problem in the pericardial segmentation produced a cut on the coronary artery path of the anterior descending artery in the medial part of the heart (see Figure 22).

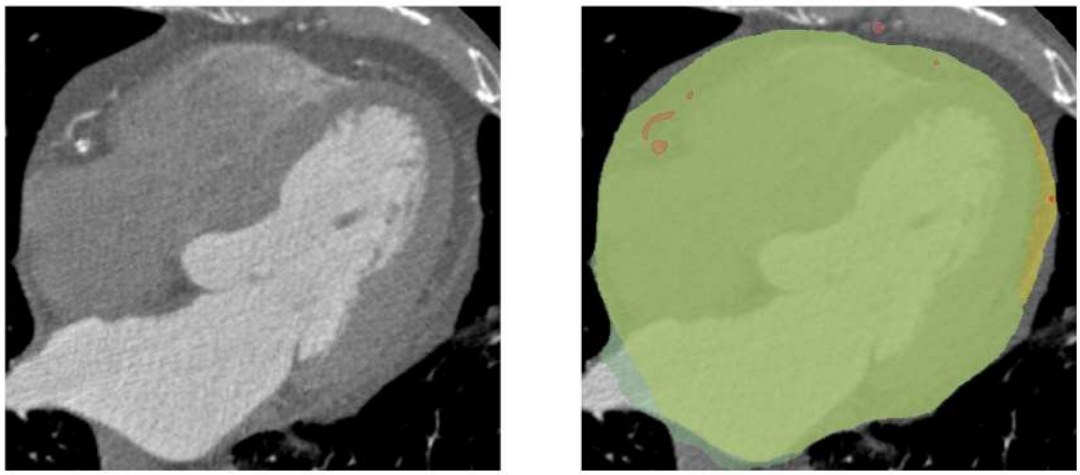


Figure 22: Transverse cut displaying an error in the pericardial segmentation. Red: automatically segmented coronary artery. Dark green: automatically segmented pericardium. Yellow: pericardial ground truth. Yellowish green: superposition of ground truth and automated segmentation.

In some other cases, some sections containing FP were produced, but the segmentation usually lied next to the edge of the pericardium, and it was within a margin in which it could not confuse the coronary arteries with other vessels external to the

pericardium. In certain cases (4 out of 11), it segmented bigger volumes external to the pericardium as part of the pericardium, such as those corresponding to the stomach, diaphragm, or liver. However, they did not usually contain vessels that could potentially be mistaken for coronary vessels. This problem was solved after selecting the largest connected component, which corresponded to the pericardial sac, and discarding the others related to other cavities.

Slices down-sampling

In order to reduce the computational efforts needed to carry out the training of the algorithms, another experimental procedure was used, which consisted of reducing the image resolution. After the first step, consisting of reducing the dimensions of each slice, a visual inspection of the cases was carried out in order to discard inadequate approximations. This was the case of the 4x4 and 8x8 down-sampling, which did not have sufficient image quality to clearly discern the coronary structures. Instead, the wider vessels were only represented by 3 or 4 pixels in each slice, while the thinner ones were not represented at all. Regarding 2x2 down-sampling, the images were clearly of inferior resolution, but the coronary structures could still be seen, so training was conducted using only that certain down-sampling. Figure 23 shows the difference in resolution between the two selected sampling methods.

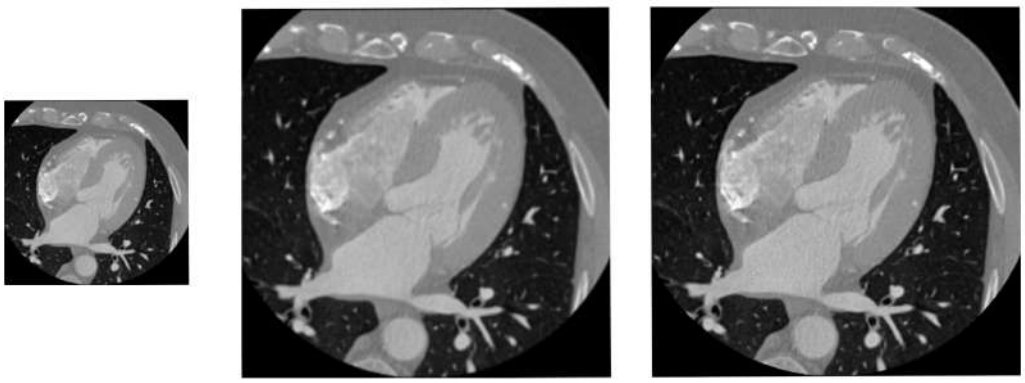


Figure 23: Comparison of the slices before and after the down-sampling. Left: slice after the 2-fold down-sampling, with 256x256 voxels. Center: increased size 2x2 down-sampled slice to match the size of the full resolution one. Right: original sized slice, with 512x512 voxels.

Training, in this case, was carried out using the same 29 training images as in previous cases but using the down-sampled version of them. The number of epochs was also kept at 500. Using this configuration, the training lasted nine days and 22 hours.

After the training, the inference was computed, and the obtained segmentations were analyzed. This study showed that the segmentations lost many important features by using this specific approach. Principally, the most distal segments of the coronary arteries (which were those with smaller caliber) were missing in most of the images. In addition, small-caliber vessels were also missing or segmented with many interruptions. Figure 24 compares an automatic segmentation with the ground truth. There, it can be seen how the diagonal coronary was not segmented in its medial and distal segments and that the anterior descending artery has many interruptions. On the other hand, the segmentation of the right coronary artery was correct.

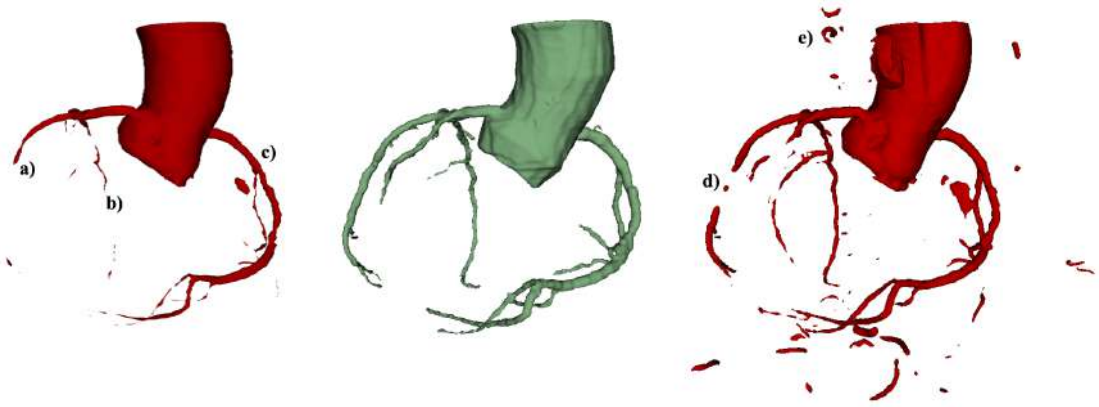


Figure 24: Comparison of the results using different segmentation approaches. Left: automatically segmented coronary artery using the slice down-sampling approach. Center: ground truth segmentation. Right: automatically segmented coronary artery using original sized images. a) Missing diagonal coronary. b) Wrongly segmented anterior descending artery. c) Correctly segmented right coronary artery. d) Segmentation stop in the diagonal artery. e) Wrongly segmented external vessels.

3.1.3 Final model

Design of the final model

After conducting the aforementioned experiments, the next step was to design the final segmentation model based on the received feedback from the experimental models studied. The methodology based on a full resolution 3D U-Net using 512x512 sized slices and embedded in the nnU-Net framework seemed correct to be used as the backbone of the model. Moreover, as shown in the experimental set-up on which the suitability of discarding the vessels external to the pericardial sac was studied, it was a good approach, too, due to its ability to actually discarding those vessels. Furthermore, on some occasions, the final coronary artery segmented volume required a selection of the largest connected component.

Final model performance

With respect to the final model, Figures 25 and 26 display an illustrative example of the studied workflow that allows segmentation of the coronary artery. Regarding the segmentation time, it takes more time than the CT Comprehensive Cardiac Analysis software, without including mesh post-processing. However, the achieved segmentation is more suitable for subsequent simulations and related applications in the case of the studied workflow.

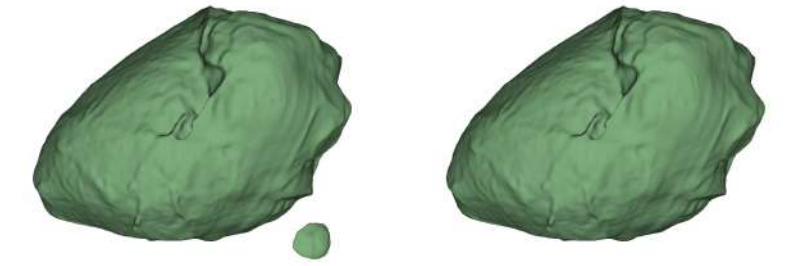


Figure 25: Example of pericardial segmentation using the final model. Left: initially segmented pericardial sac. Right: final pericardial sac after discarding the non-largest connected components.

Figure 26: Example of coronary segmentation using the final model. Left: initially segmented coronary artery. Center: intersection of the initially segmented coronary artery and the final pericardial sac. Right: final coronary artery.

3.2 Proof-of-concept

After the design of the workflow, its performance was evaluated for an example, as a proof of concept, including all the steps explained before.

3.2.1 Mesh generation and post-processing

First, regarding the segmentation of the structures, Figure 27 shows the obtained segmentation for the pericardial cavity with and without post-processing. In addition, Figure 28 displays the achieved segmentation for the coronary artery in three steps: raw segmentation, the intersection of the raw segmentation with the final pericardium segmentation, and the segmentation after discarding the non-largest connected components. The latter is the segmentation that would be used from that moment on.

After the post-processing of the binary mask and its validation by a cardiac imaging expert physician, the mesh was created. Figure 29a shows a detail of the mesh, in which it can be seen a shortcut between the circumflex artery and the aortic root. In addition, the same image shows the smoothness of the mesh and the correct segmentation of the aortic root, unlike the one achieved by the CT Comprehensive Cardiac Analysis tool by Philips, shown in Figure 29b, which was also retrieved for comparison purposes.

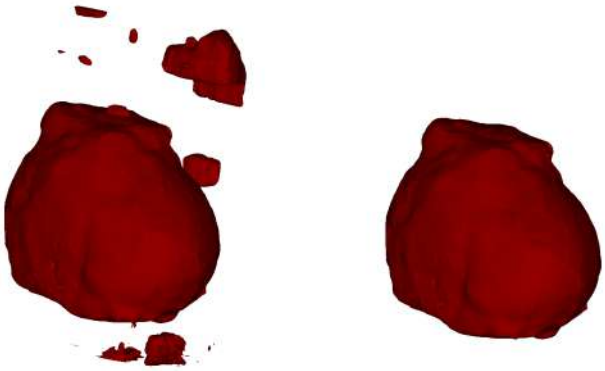


Figure 27: Pericardial segmentation for the proof-of-concept. Left: initially segmented pericardial sac. Right: post-processed pericardial sac.

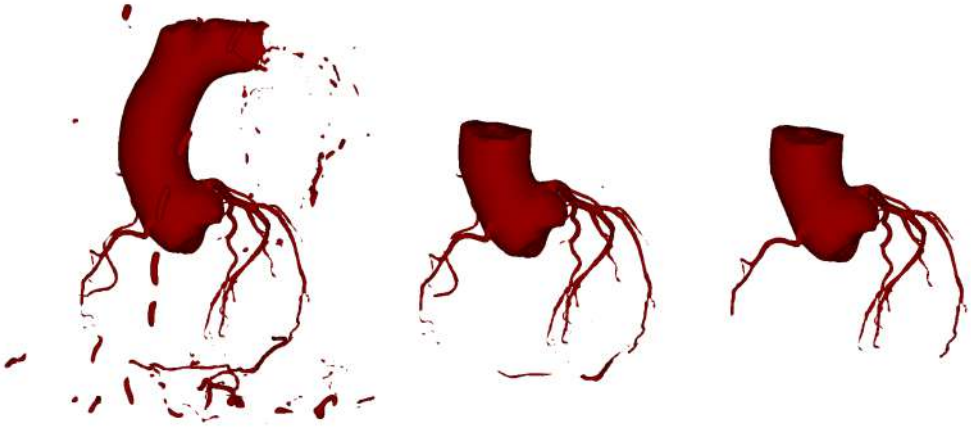
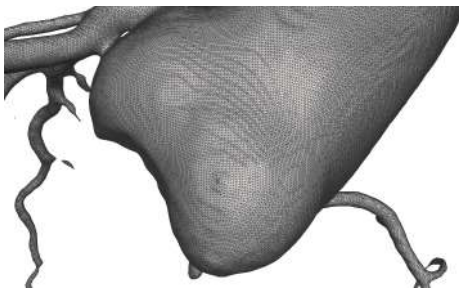
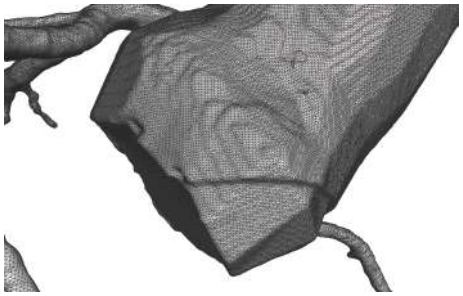


Figure 28: Coronary segmentation for the proof-of-concept. Left: initially segmented coronary artery. Center: intersection of the initially segmented coronary artery and the final pericardial sac. Right: final coronary artery.



(a) Segmentation obtained using the nnU-Net-based workflow.



(b) Segmentation obtained using the CT Comprehensive Cardiac Analysis tool.

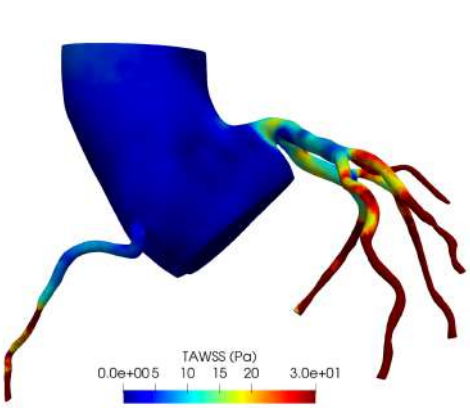
Figure 29: Comparison of the segmentation quality, specially in the aortic root, and mesh smoothness between nnU-Net-based algorithm and CT Comprehensive Cardiac Analysis tool.

3.2.2 Visualization of the results

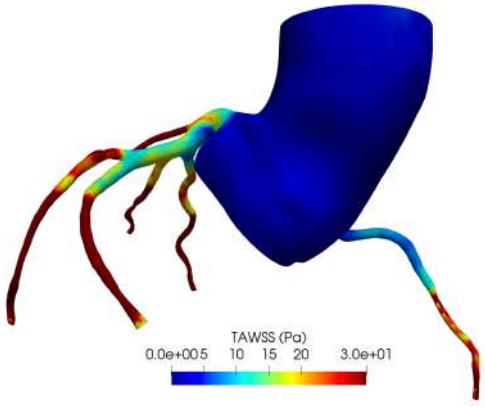
Using the created mesh, the CFD simulation was computed on a 4-core CPU, taking 2 hours to complete. Then, the next step was to visualize the resulting hemodynamic parameters. The first visualization consisted of displaying the WSS values of the model wall. Since WSS is a dynamic factor, which depends on the phase of the cardiac cycle analyzed, the time-averaged WSS (TAWSS) was displayed to get an idea of the average WSS during the entire cardiac cycle. Figure 30 shows the simulated TAWSS in the coronary tree. Moreover, the streamlines showing the direction and velocity of the simulated blood flow were also computed. Figure 31 displays them in two different stages of the cardiac cycle: in systole (Figure 31a) and diastole (Figure 31b). Their behaviour in both moments agree with physiological values.

3D printing

Pending results.

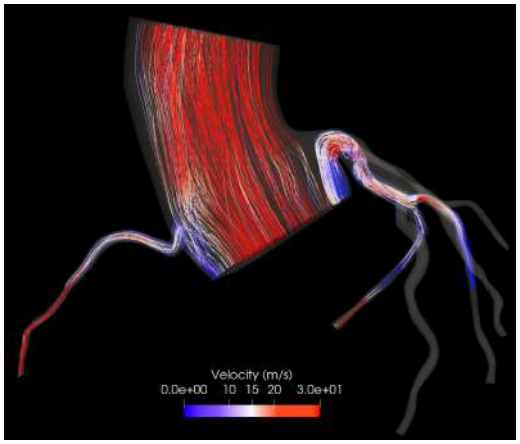


(a) Frontal view of the coronary tree colored by the TAWSS.

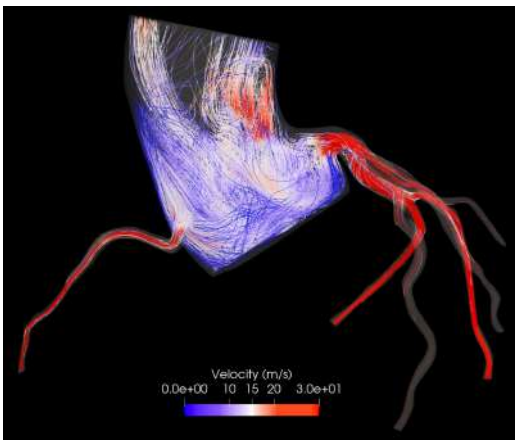


(b) Posterior view of the coronary tree colored by the TAWSS.

Figure 30: Frontal and posterior views of the coronary tree colored by the simulated TAWSS. TAWSS: Time-Averaged Wall Shear Stress.



(a) Systole. The aortic valve is open, so blood flows from the aortic inlet to the ascending aorta.



(b) Diastole. The aortic valve is closed, so blood flows into the coronary arteries forming a characteristic funnel-shaped structure.

Figure 31: Flow lines within the coronary arteries in systole and diastole.

Chapter 4

Discussion

In cardiac imaging units, image segmentation is the major bottleneck. It requires much time to be computed, apart from the time used in extracting the clinically essential information and tabulating it. This time spent by physicians is time not spent attending more patients or diagnosing based on imaging data. For this, the development and implementation of a workflow allowing the fast processing of multi-modal data is fundamental.

We propose to do precisely so in this thesis. In this workflow, all steps from data acquisition to fluid simulations have been studied. Some state-of-the-art algorithms are included within it, such as U-Net, which is implemented using the nnU-Net framework, producing high-quality results for segmentations.

4.1 Coronary segmentation model

Using the DL-based segmentation algorithms, the high quality of the obtained segmentations is one of the strongest points of this work. Having a well-done segmentation in a fully automated way is fundamental to simplify this labor, which is conducted in a semi-automatic manner nowadays in cardiac imaging units. The only manual task of the physician would be, in this case, to validate the segmentations and to correct possible errors. When comparing the performance of the segmenta-

tion methodology developed in this work with that of CT Comprehensive Cardiac Analysis software, it can be found that the segmentation time increase when using the former method. However, the quality of the segmentation and the smoothness of the mesh do as well. For this reason, it can be therefore inferred that the total time of segmentation and adaptation of the mesh to be used is significantly lower by using the methodology developed in the current study. This way, the integration of on-site CFD studies would be more user-friendly following this workflow. In addition, these CFD simulations could be validated by means of *in vitro* test-rigs using 3D printed coronary phantoms [55] [56].

Nevertheless, this work has revealed some pitfalls, whose solving would significantly improve the performance of the algorithm. In the first place, segmentation performance has not been assessed through any quantitative metrics but the qualitative criteria of professionals. The Dice coefficient was calculated, but because of the coarse segmentation of the aorta in the ground truth, which produced FP and FN more important in volume than the errors in the coronary segmentation, we preferred to follow the advice of the cardiac imaging experts. However, this lack of objective metrics could therefore pose a problem when generalizing. Furthermore, although the training has been carried out with images obtained by more than one vendor, the validation of these images has been performed by only one professional. If its use has to be widespread, the opinion of other experts should be considered. In addition, the training set of the method should contain images from multiple centers and different vendors and should be comprised of many more images than the ones used in this work. This would ensure the presence of coronary arteries with rare anatomies in the training set. Finally, the original goal of this study was not completely fulfilled since inference was not carried out in Hospital de Sant Pau but at UPF HPC Cluster. It also posed a problem regarding the limited computing power available in that cluster, which did not permit faster or more advanced training. Thus, other deployment options, such as on-site or cloud-based implementation, were explored.

Regarding the segmentation algorithms *per se*, the use of a nnU-Net as the backbone

is not completely new in coronary arterial lumen segmentation since some of the teams participating in the ASOCA challenge used that approach, achieving results (Dice score of 0.867 for the best qualifying team). However, the segmentation model presented in this work is not comparable to those of ASOCA, because of the lack of segmentation of the aortic root in the challenge. Still, it is interesting to study the incorporation of segmentation methodologies presented in challenges to the medical world.

4.2 Data management and deployment

As seen before, incorporating this workflow into the current clinical practice would be largely beneficial, especially for saving time and reducing intra-observer variability. Besides, a proof-of-concept has been conducted, demonstrating its viability and convenience. Nonetheless, the connection among the different modules and software has been conducted manually, so it has not been completely included in the daily practice. Therefore, this is another limitation of this work since it has not been possible to test its implementation within the usual workflow in the hospital.

However, the deployment of the workflow leads to other challenges, including logistical problems. For example, due to the high demand for resources required by these novel algorithms, we should consider setting up structures designed almost exclusively for their training. Moreover, issues such as data collection and anonymization are fundamental. It is also crucial to integrate these algorithms with the hospital's PACS system to obtain the segmentations automatically right after image acquisition, performing in the background, and with minimal human intervention. To this end, it is important to contact the manufacturers of these PACS systems and try to integrate these methods whenever possible.

Previously, some deployment options have been discussed, each of them presenting some pros and cons. For example, the approach related to the creation of an AI unit within the hospital seems the most natural because, in this way, the data would not leave the hospital, and the center would be responsible for determining

the exact computational power that would be needed. This method might seem the most suitable for centres of sufficient size, but it is a very expensive option, as maintenance costs for servers and AI infrastructure would be significantly high.

Therefore, the second approach, which would consist of bringing together several centers that share certain characteristics with a university or a research center, could solve this problem. The larger centers would collaborate by providing the most data and funding, while the smaller ones would expand the geographic scope of these algorithms. In addition, the presence of researchers external to the hospitals could help in achieving a more multidisciplinary insight. However, this method may raise some problems related to possible imbalance between centers or data protection issues, which hinder the data exchange process. Here, federated learning options can be explored, using datasets distributed across multiple centers while preventing data leakage [57].

Finally, the approach based on cloud services seems fair because only the necessary resources would be rented and for the strictly necessary time. However, having a server located in an unknown location and relying on an external agency for commercial purposes could be problematic. In addition, patient data privacy issues would require an even more arduous process of patient data anonymization.

4.3 Other potential applications

One of the most interesting features of this work, as it stands, has been that, during the design and testing of the experimental setups, other potential applications were discovered. In particular, these relate to the neural network designed to segment the pericardium. This network has been used to provide the coronary arteries with an anatomical framework to discard external vessels. However, being able to perform fully automatic segmentation of the pericardium is an important starting point for several applications.

First, the pericardium serves as an envelope for the coronary arteries and the epicardial fat, which is one of the most important computational markers of CAD,

highly related to high-risk plaque and atherosclerosis [58], [59]. This fat is defined as the one located within the pericardial sac and has radiodensity values between -150 and -30 HU. Thus, the number of voxels in the pericardium with these ranges of radiodensity can be determined, and the epicardial fat can be quantitatively assessed. Moreover, the segmentation of this type of fat can be integrated within Non-Fluoroscopic Electroanatomical Mapping (CARTO[1]) systems to provide these maps with an anatomical reference for the fat.

Theoretically, this is presented as a logical and novel idea. However, this has not been tested, so studies should be carried out in this regard, in order to assess its feasibility and to explore its implications and implementation difficulties.

[1]https://www.jnjmedicaldevices.com/en-US/product-family/3d-navigation

Chapter 5

Conclusions

Finally, to synthesize all the information presented, the present work has introduced an AI-based complete workflow for the processing of multi-modality data in a cardiac imaging unit of a hospital. A use case, which was the segmentation of coronary arteries from CCT images, was also surveyed and implemented at the *Hospital de Sant Pau*. All the steps, from the training data set to the deployment options, were analyzed and the results of simulations performed after segmentation presented. Furthermore, these simulations have provided physiologically reasonable results.

One of the most important conclusions of this work is that the feasibility of the project is high, provided that its limitations can be overcome. If successful, i.e., by increasing the training data set and retraining the networks with feedback from expert annotators, it is possible to integrate it into day-to-day clinical practice. However, the fact that the segmentation algorithm is able to generalize well even without a very precise ground truth segmentation is very encouraging.

The implementation of this workflow is another problem, as it has not been tested on-site. Thus, a detailed analysis of it would be necessary to make this type of workflow possible in the short-medium run.

The next step after implementing these advanced segmentation algorithms would consist of automating the subsequent processes, especially in obtaining the computa-

tional markers of disease necessary for diagnosis. Thus, the workload of medical professionals would decrease, as they would only be responsible for the post-processing and validation of the segmentation. Therefore, they would be able to devote more time to studying these markers and the diagnosis of patients.

List of Figures

1 Pipeline showing the steps of the cardiac imaging workflow. 3
2 Artificial intelligence impact in the various steps of the cardiac imaging pipeline. 5
3 Example U-Net architecture. 6
4 Representation of the nnU-net configuration and description of the hyper-parameters. 7
5 Anatomy of the coronary tree. 8
6 Utility of cardiac computed tomography in coronary artery disease. . 11
7 Summary of the thesis workflow adapted to the coronary artery segmentation task. 16
8 Transverse cut of a cardiac computed tomography volume highlighting the coronary arterial lumen. 23
9 Workflow for the dataset creation. 25
10 Disposition of the data within the folders to allow its compatibility within the nnU-Net framework. 26
11 Full resolution 3D U-Net architecture trained on the nnU-Net framework and used for the coronary dataset. 28
12 Example slice of a cardiac computed tomography volume displaying two types of arteries. 29
13 Coronal plane of a cardiac computed tomography. 30
14 Manually annotated slices, used in multi-slice editing. 31
15 Example folder structure for the two tasks to make data compatible with nnU-Net framework. 32
16 General workflow for the segmentation model. 33

17 Example of automatic segmentation before and after manually adding the anterior descending artery. 38
18 Example of ground truth included into the dataset. 39
19 Automated segmentation compared to the ground truth. 39
20 Classification of the errors of the automatic segmentation based on their importance. 40
21 Semi-automatically segmented pericardium. 41
22 Transverse cut displaying an error in the pericardial segmentation. . . 42
23 Comparison of the slices before and after the down-sampling. 43
24 Comparison of the results using different segmentation approaches. . 44
25 Example of pericardial segmentation using the final model. 45
26 Example of coronary segmentation using the final model. 46
27 Pericardial segmentation for the proof-of-concept. 47
28 Coronary segmentation for the proof-of-concept. 47
29 Comparison of the segmentation quality and mesh smoothness between nnU-Net-based algorithm and CT Comprehensive Cardiac Analysis tool. 47
30 Frontal and posterior views of the coronary tree colored by the simulated Time-averaged wall shear stress. 49
31 Flow lines within the coronary arteries in systole and diastole. 49

A1 Example in which a complete artery was not segmented. 72
A2 Example in which there was a stop in the segmentation of a principal vessel. 72
A3 Example in which an anatomy-related problem affected the segmentation. 73
A4 Example in which a heart cavity was segmented as coronary artery. . 73
A5 Example of a segmented vein within the pericardium. 74
A6 Example in which a distal segment was incorrectly segmented. 74

B1 Mesh obtained after the edition of the distal segments of the coronary arteries. 76

B2 Cardiac computed tomography image highlighting the contour of the measured right ventricular outflow tract area. 77
B3 Pulsed-Wave Doppler mode used to compute the velocity integral over time. 78
B4 Volumetric mesh displaying the external faces of the 3D discretization. 79
B5 Aortic inflow waveform. 80
B6 Circuits depicting the boundary conditions for the aortic and the coronary outflows. 80

List of Tables

1 Information about the multi-vendor cardiac computed tomography dataset. 24

2 Presentation of the most common segmentation problems encountered in the first coronary segmentation model. 41

A1 Description of the problems, the positive aspects and other considerations for each case in the test set after the first coronary segmentation model. 70

A2 Analysis of the prevalence and the importance of different problems encountered in the first coronary segmentation model. In addition, discussion of the causes and potential solutions to them. 71

Bibliography

[1] Chen, C. *et al.* Deep Learning for Cardiac Image Segmentation: A Review. *Frontiers in Cardiovascular Medicine* **7** (2020). 1911.03723.

[2] Tavakoli, V. & Amini, A. A. A survey of shaped-based registration and segmentation techniques for cardiac images. *Computer Vision and Image Understanding* **117**, 966–989 (2013). URL https://doi.org/10.1016/j.cviu.2012.11.017.

[3] Badano, L. Artificial Intelligence and Cardiovascular Imaging. A win-win Combination. *The Anatolian Journal of Cardiology* 214–223 (2020).

[4] Loncaric, F., Camara, O., Piella, G. & Bijnens, B. Integration of artificial intelligence into clinical patient management: focus on cardiac imaging. *Revista Española de Cardiología (English Edition)* **74**, 72–80 (2021). URL https://doi.org/10.1016/j.rec.2020.07.003.

[5] Sanchez-Martinez, S. *et al.* Machine learning for clinical decision-making: Challenges and opportunities (2019). URL https://doi.org/10.20944/preprints201911.0278.v1.

[6] Kelly, C. J., Karthikesalingam, A., Suleyman, M., Corrado, G. & King, D. Key challenges for delivering clinical impact with artificial intelligence. *BMC Medicine* **17**, 1–9 (2019).

[7] Mechelli, A. & Vieira, S. From models to tools: clinical translation of machine learning studies in psychosis. *npj Schizophrenia* **6** (2020). URL http://dx.doi.org/10.1038/s41537-020-0094-8.

[8] Singh, R. P., Hom, G. L., Abramoff, M. D., Campbell, J. P. & Chiang, M. F. Current challenges and barriers to real-world artificial intelligence adoption for the healthcare system, provider, and the patient. *Translational Vision Science and Technology* **9**, 1–6 (2020).

[9] Rong, G., Mendez, A., Bou Assi, E., Zhao, B. & Sawan, M. Artificial Intelligence in Healthcare: Review and Prediction Case Studies. *Engineering* **6**, 291–301 (2020). URL `https://doi.org/10.1016/j.eng.2019.08.015`.

[10] Santini, G. *et al.* Synthetic contrast enhancement in cardiac CT with Deep Learning 1–8 (2018). URL `http://arxiv.org/abs/1807.01779`. `1807.01779`.

[11] van Hamersvelt, R. W. *et al.* Deep learning analysis of left ventricular myocardium in CT angiographic intermediate-degree coronary stenosis improves the diagnostic accuracy for identification of functionally significant stenosis. *European radiology* **29**, 2350–2359 (2019).

[12] Wolterink, J. M. *et al.* Automatic coronary artery calcium scoring in cardiac CT angiography using paired convolutional neural networks. *Medical image analysis* **34**, 123–136 (2016).

[13] Maier-Hein, L. *et al.* Why rankings of biomedical image analysis competitions should be interpreted with care. *Nature Communications* **9** (2018). URL `https://doi.org/10.1038/s41467-018-07619-7`.

[14] SN, K., A, L. F., S, M., H, A. K. & P, S. V. A voyage on medical image segmentation algorithms. *Biomedical Research* (2018). URL `https://doi.org/10.4066/biomedicalresearch.29-16-1785`.

[15] Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Lecture Notes in Computer Science*, 234–241 (Springer International Publishing, 2015). URL `https://doi.org/10.1007/978-3-319-24574-4_28`.

[16] Isensee, F. & Maier-Hein, K. An attempt at beating the 3d u-net. *ArXiv* **abs/1908.02182** (2019).

[17] Isensee, F., Jaeger, P. F., Kohl, S. A. A., Petersen, J. & Maier-Hein, K. H. nnU-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods* (2020). URL `https://doi.org/10.1038/s41592-020-01008-z`.

[18] Ma, J. Cutting-edge 3D Medical Image Segmentation Methods in 2020: Are Happy Families All Alike? *Preprints* 1–13 (2021). URL `http://arxiv.org/abs/2101.00232`. `2101.00232`.

[19] Lynch, P. J. Anatomy of the coronary tree (2010). URL `https://commons.wikimedia.org/w/index.php?curid=9967381`.

[20] Ference, B. A., Ginsberg, H. N., Graham, I. & et al. Low-density lipoproteins cause atherosclerotic cardiovascular disease. 1. Evidence from genetic, epidemiologic, and clinical studies. A consensus statement fromthe European Atherosclerosis Society Consensus Panel. *European Heart Journal* **38**, 2459–2472 (2017).

[21] Bentzon, J. F., Otsuka, F., Virmani, R. & Falk, E. Mechanisms of plaque formation and rupture. *Circulation Research* **114**, 1852–1866 (2014).

[22] Kannel, W. B. CHD risk factors: A Framingham study update. *Hospital Practice* **25**, 119–130 (1990).

[23] Lusis, A. J. Atherosclerosis. *Nature* **407**, 233–241 (2000).

[24] Ghanem, A. M. *et al.* Automatic Coronary Wall and Atherosclerotic Plaque Segmentation from 3D Coronary CT Angiography. *Scientific Reports* **9**, 1–13 (2019).

[25] Levy, D. Combating the epidemic of heart disease. *JAMA - Journal of the American Medical Association* **308**, 2624–2625 (2012).

[26] Murray, C. J. & Lopez, A. D. Measuring the Global Burden of Disease. *New England Journal of Medicine* **369**, 448–457 (2013).

[27] Ferreira-González, I. The Epidemiology of Coronary Heart Disease. *Revista Española de Cardiología (English Edition)* **67**, 139–144 (2014).

[28] Naghavi, M., Wang, H., Lozano, R. & et. al. Global, regional, and national age-sex specific all-cause and cause-specific mortality for 240 causes of death, 1990-2013: A systematic analysis for the Global Burden of Disease Study 2013. *The Lancet* **385**, 117–171 (2015). URL `http://dx.doi.org/10.1016/S0140-6736(14)61682-2`.

[29] Gorenoi, V., Schönermark, M. P. & Hagen, A. CT coronary angiography vs. invasive coronary angiography in CHD. *GMS Health Technology Assessment* **8**, Doc02 (2012).

[30] Kolossváry, M., Szilveszter, B., Merkely, B. & Maurovich-Horvat, P. Plaque imaging with CT-A comprehensive review on coronary CT angiography based risk assessment. *Cardiovascular Diagnosis and Therapy* **7**, 489–506 (2017).

[31] Pijls, N. H. *et al.* Measurement of Fractional Flow Reserve to Assess the Functional Severity of Coronary-Artery Stenoses. *New England Journal of Medicine* **334**, 1703–1708 (1996).

[32] Britton, K. A. & Fox, C. S. Perivascular adipose tissue and vascular disease. *Clinical Lipidology* **6**, 79–91 (2011).

[33] Abdelrahman, K. M. *et al.* Coronary computed tomography angiography from clinical uses to emerging technologies. *Journal of the American College of Cardiology* **76**, 1226–1243 (2020). URL `https://doi.org/10.1016/j.jacc.2020.06.076`.

[34] Liu, J. *et al.* A Vessel-Focused 3D Convolutional Network for Automatic Segmentation and Classification of Coronary Artery Plaques in Cardiac CTA. In Pop, M. *et al.* (eds.) *Statistical Atlases and Computational Models of the Heart. Atrial Segmentation and LV Quantification Challenges*, 131–141 (Springer International Publishing, Cham, 2019).

[35] Huang, W. *et al.* Coronary artery segmentation by deep learning neural networks on computed tomographic coronarangiographic images. In *2018 40th*

Annual International Conference of the IEEE Engineering in Medicine and BiologSociety (EMBC) (IEEE, 2018). URL https://doi.org/10.1109/embc.2018.8512328.

[36] Benton, S. M. *et al.* Noninvasive derivation of fractional flow reserve from coronary computed tomographic angiography. *Journal of Thoracic Imaging* **33**, 88–96 (2018). URL https://doi.org/10.1097/rti.0000000000000289.

[37] Röther, J. *et al.* Comparison of invasively measured FFR with FFR derived from coronary CT angiography for detection of lesion-specific ischemia: Results from a PC-based prototype algorithm. *Journal of Cardiovascular Computed Tomography* **12**, 101–107 (2018). URL https://doi.org/10.1016/j.jcct.2018.01.012.

[38] Peiffer, V., Sherwin, S. J. & Weinberg, P. D. Does low and oscillatory wall shear stress correlate spatially with early atherosclerosis? a systematic review. *Cardiovascular Research* **99**, 242–250 (2013). URL https://doi.org/10.1093/cvr/cvt044.

[39] Arzani, A. Coronary artery plaque growth: A two-way coupled shear stress–driven model. *International Journal for Numerical Methods in Biomedical Engineering* **36** (2019). URL https://doi.org/10.1002/cnm.3293.

[40] Cui, J. *et al.* Fully-automatic segmentation of coronary artery using growing algorithm. *Journal of X-ray science and technology* (2020).

[41] Kong, B. *et al.* Learning tree-structured representation for 3d coronary artery segmentation. *Computerized Medical Imaging and Graphics* **80**, 101688 (2020). URL https://doi.org/10.1016/j.compmedimag.2019.101688.

[42] Hampe, N., Wolterink, J. M., van Velzen, S. G. M., Leiner, T. & Išgum, I. Machine Learning for Assessment of Coronary Artery Disease in Cardiac CT: A Survey. *Frontiers in Cardiovascular Medicine* **6**, 172 (2019). URL https://www.frontiersin.org/article/10.3389/fcvm.2019.00172.

[43] Zhao, F. *et al.* An automatic multi-class coronary atherosclerosis plaque detection and classification framework. *Medical & Biological Engineering & Computing* **57**, 245–257 (2019). URL https://doi.org/10.1007/s11517-018-1880-6.

[44] Zuluaga, M. A., Hush, D., Leyton, E. J. F. D., Hoyos, M. H. & Orkisz, M. Learning from only positive and unlabeled data to detect lesions in vascular ct images. In *Proceedings of the 14th International Conference on Medical Image Computing and Computer-Assisted Intervention - Volume Part III*, MICCAI'11, 9–16 (Springer-Verlag, Berlin, Heidelberg, 2011).

[45] Cano-Espinosa, C., González, G., Washko, G. R., Cazorla, M. & Estépar, R. S. J. Automated Agatston Score Computation in non-ECG Gated CT Scans Using Deep Learning. *Proceedings of SPIE–the International Society for Optical Engineering* **10574** (2018).

[46] de Vos, B. D. *et al.* Direct Automatic Coronary Calcium Scoring in Cardiac and Chest CT. *IEEE transactions on medical imaging* **38**, 2127–2138 (2019).

[47] Iacobellis, G. *Coronary Artery Disease and Epicardial Adipose Tissue*, 77–90 (Springer International Publishing, Cham, 2020). URL https://doi.org/10.1007/978-3-030-40570-0{_}8.

[48] Oksuz, I. *et al.* Automatic CNN-based detection of cardiac MR motion artefacts using k-space data augmentation and curriculum learning. *Medical Image Analysis* **55**, 136–147 (2019). URL https://doi.org/10.1016/j.media.2019.04.009.

[49] Sommer, K. N. *et al.* 3d printed cardiovascular patient specific phantoms used for clinical validation of a CT-derived FFR diagnostic software. In Gimi, B. & Krol, A. (eds.) *Medical Imaging 2018: Biomedical Applications in Molecular, Structural, and Functional Imaging* (SPIE, 2018). URL https://doi.org/10.1117/12.2292736.

[50] Yan, W. *et al.* MRI manufacturer shift and adaptation: Increasing the generalizability of deep learning segmentation for MR images acquired with different scanners. *Radiology: Artificial Intelligence* **2**, e190195 (2020). URL `https://doi.org/10.1148/ryai.2020190195`.

[51] Fedorov, A. *et al.* 3d slicer as an image computing platform for the quantitative imaging network. *Magnetic Resonance Imaging* **30**, 1323–1341 (2012). URL `https://doi.org/10.1016/j.mri.2012.05.001`.

[52] Kluyver, T. *et al.* Jupyter notebooks - a publishing format for reproducible computational workflows. In Loizides, F. & Scmidt, B. (eds.) *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, 87–90 (IOS Press, Netherlands, 2016). URL `https://eprints.soton.ac.uk/403913/`.

[53] Ulyanov, D., Vedaldi, A. & Lempitsky, V. Instance normalization: The missing ingredient for fast stylization (2017). `1607.08022`.

[54] van der Walt, S. *et al.* scikit-image: image processing in python. *PeerJ* **2**, e453 (2014). URL `https://doi.org/10.7717/peerj.453`.

[55] Shepard, L. M. *et al.* Initial evaluation of three-dimensionally printed patient-specific coronary phantoms for CT-FFR software validation. *Journal of Medical Imaging* **6**, 1 (2019). URL `https://doi.org/10.1117/1.jmi.6.2.021603`.

[56] Morris, P. D. *et al.* A novel method for measuring absolute coronary blood flow and microvascular resistance in patients with ischaemic heart disease. *Cardiovascular Research* **117**, 1567–1577 (2020). URL `https://doi.org/10.1093/cvr/cvaa220`.

[57] Yang, Q., Liu, Y., Chen, T. & Tong, Y. Federated machine learning. *ACM Transactions on Intelligent Systems and Technology* **10**, 1–19 (2019). URL `https://doi.org/10.1145/3298981`.

[58] Nerlekar, N. *et al.* Association of epicardial adipose tissue and high-risk plaque characteristics: A systematic review and meta-analysis. *Journal of the Ameri-*

can Heart Association **6** (2017). URL https://doi.org/10.1161/jaha.117.006379.

[59] Goeller, M. *et al.* Epicardial adipose tissue density and volume are related to subclinical atherosclerosis, inflammation and major adverse cardiac events in asymptomatic subjects. *Journal of Cardiovascular Computed Tomography* **12**, 67–73 (2018). URL https://doi.org/10.1016/j.jcct.2017.11.007.

[60] Si, H. TetGen, a delaunay-based quality tetrahedral mesh generator. *ACM Transactions on Mathematical Software* **41**, 1–36 (2015). URL https://doi.org/10.1145/2629697.

[61] Kim, H. J. *et al.* Patient-specific modeling of blood flow and pressure in human coronary arteries. *Annals of Biomedical Engineering* **38**, 3195–3209 (2010). URL https://doi.org/10.1007/s10439-010-0083-6.

[62] Landes, G. Einige untersuchungen an elektrischen analogieschaltungen zum kreitslaufsystem. *Zeitschrift fur Biologie* **101**, 418–429 (1943).

Appendix A

First Appendix

A.1 Analysis of the first segmentation algorithm.

This chapter includes an analysis of the first version of the segmentation algorithm, i. e., that consisting of training a full-resolution 3D-UNet using the nnU-Net framework and a training set size of 29 images during 500 epochs.

First, Table A1 shows a description of the problems, the positive aspects, and other considerations for each case in the test set after carrying out the inference for the first version of the algorithm. On the other hand, in Table A2, analysis for each problem is presented. It contains detailed information on the importance and prevalence of each of them, combined with a discussion of the causes and potential solutions. In addition, some illustrative images for each case are additionally presented.

Table A1: Description of the problems, the positive aspects and other considerations for each case in the test set after the first coronary segmentation model.

Case ID	Problems	Positive aspects to highlight	Other considerations
1	Atrial diverticulum erroneously segmented. One segmented vein	Overall good quality segmentation	Image presenting a left atrial diverticulum (tubular structure, with contrast, rare) Granulated image, not very smooth
2	Segment with myocardial bridging not segmented (distal and proximal segments were correctly segmented) Interruption of the circumflex artery Short diagonal artery not segmented	Taking into account all the difficulties of the case, good segmentation.	Difficult to segment study Poor quality Motion during acquisition Poor signal-to-noise ratio Grainy image Myocardial bridging over the anterior descending artery
3	Segment with myocardial bridging not segmented	Obstructive plate correctly interrupted luminal segmentation of circumflex artery	Myocardial bridging over the anterior descending artery
8	Erroneous segmentation of the left atrial appendage Acute marginal artery not segmented		
10	Three stops: circumflex artery, diagonal artery and anterior descending artery	Very precise aortic segmentation Low noise external to pericardium	
11	Pulmonary arteries identified as coronary ones Conal artery missing		There is contrast in the right ventricle
12	Problems in the distal segments of the thinner arteries The marginal artery and the distal segment of the anterior descending artery wrre missing Part of the right ventricle was segmented		
13	Posterior descending artery not segmented Diagonal artery not segmented	Correct stops in anterior descending artery	Cardiac computed tomography acquisition in motion, so it was challenging No visible drop in contrast nor wall thickening
14	Segmentation of intra-pericardial veins	Lumen well segmented, same as ground truth Segmentation better than that done by available software	Late study, presents contrast in veins
15	Error in segmenting a (large) diagonal artery Coronary artery abnormality erroneously segmented		Case with coronary artery abnormality Rare case
16	Marginal branch of the right coronary artery missing Vessels near to the atrium were segmented.	Some vessels not identified as coronary arteries in the ground truth were correctly segmented	

Error explanation	No. of cases (n=11)	Probability of recurrence	Importance	(Possible) causes	Potential solutions
External vessels to the pericardium were segmented	11	Very high	Low. Easy to solve, easy to discern for an expert.	Presence of tubular structures with contrast within the cardiac computed tomography volume	Pericardial sac segmentation to discard external structures
Lack of segmentation of complete minor vessels: diagonal, acute marginal of the right ventricle, conal, posterior descending, marginal of the right coronary, marginal anterior descending. Figure A1	6	High	Medium-high. Depending on the importance of the particular vessel.	Incomplete ground truth, some vessels may be missing. Combination between small caliber, low spatial resolution and high image noise.	Increase the size of the training set. Careful creation of the ground truth segmentations. Artificially enlarge the lumen in some specific segments in the ground truth.
It stopped segmenting in some locations of main vessels. It was not caused by a decrease in contrast (probably due to plaque) nor by a visible thickening of the arterial wall: in circumflex artery (x2) and in diagonal (x2). Figure A2	4	High	High. Disruption due to segmentation failure could be confused with disruption due to severe plaque/calcium stenosis.	The ground truth probably contains images in which the segmentation in a vessel stops when detecting plaque.	Increase the size of the training set. Do a more accurate segmentation of the ground truth in all cases; just in case there is a problem in this regard.
Rare anatomy-dependent problems: myocardial bridging (x2), atrial diverticulum (x1), coronary artery abnormality (x1) Figure A3.	4	Anatomy-dependent	High in the cases when segmentation is interrupted or some artery is missing. Medium-low in the cases when a new structure is added.	Low number of similar cases in the training set.	Ensure that a significant number of cases with relatively frequent anomalies are incorporated into the training set.
Segmentation of part of some heart cavities as coronary arteries: left atrial appendage (x1) and right ventricle (x1). Figure A4.	2	High	Low. These segments do not correspond to any structure that could be confused with a coronary artery.	Closeness of these structures to the coronary arteries. Late studies.	Increase training set size. Potentially add an algorithm to segment other cardiac chambers, allowing it to identify them individually and reduce these types of errors.
Segmented veins within the pericardium. Figure A5.	2	Acquisition-dependent	Low. A professional will automatically discard this vessel because of its anatomical location.	Late studies, the contrast has time to penetrate the veins and to be confused with arteries.	Ensure that the studies are of the best possible quality. Connected component analysis to exclude this type of non-connected vessels.
Errors when segmenting the most distal segment of certain small caliber vessels. Figure A6.	1	High	Low, these segments were generally not considered to be as important.	Lack of examples in the training set.	Increase the size of the training set. Pay special attention to these segments.

Table A2: Analysis of the prevalence and the importance of different problems encountered in the first coronary segmentation model. In addition, discussion of the causes and potential solutions to them.

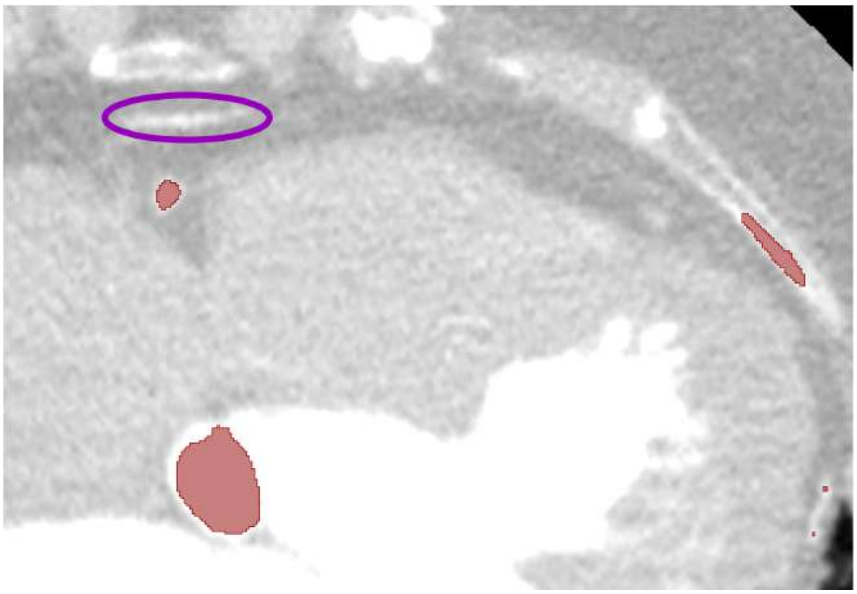


Figure A1: Example in which a complete artery was not segmented. In this case, an acute marginal branch of the right coronary artery. Purple circle: location of that artery in a transverse cut of the cardiac computed tomography image.

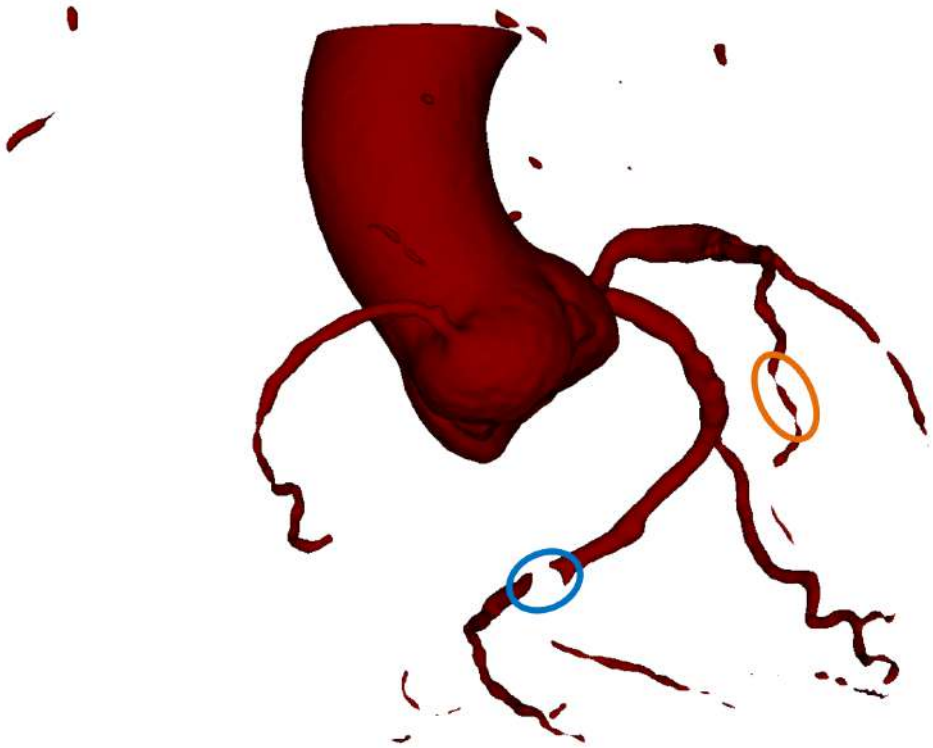


Figure A2: Example in which there was a stop in the segmentation of a principal vessel. Blue circle: location of the incorrect segmentation. Orange circle: example of good segmentation of a stenosis, validated by a cardiac imaging expert.

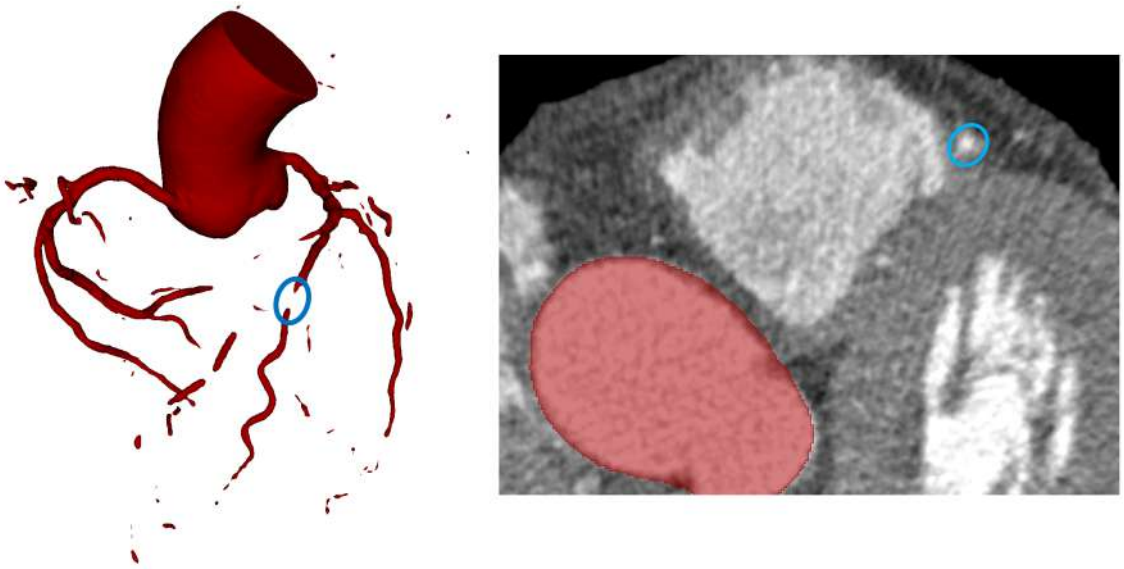


Figure A3: Example in which an anatomy-related problem affected the segmentation. In this case, a segment was undergoing myocardial bridging. Blue circles: location of the missing segment in the 3D reconstruction of the segmentation (left) and in a transverse cut of the cardiac computed tomography image (right).

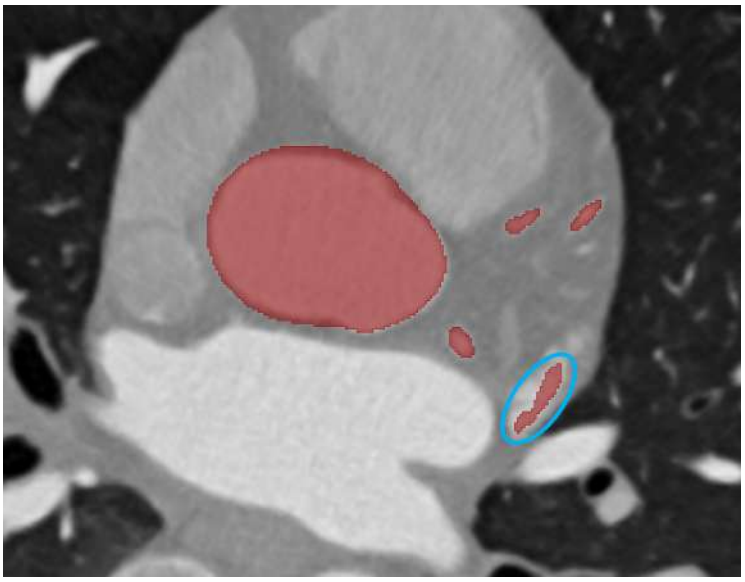


Figure A4: Example in which a heart cavity was segmented as coronary artery. In this case, this cavity was the left atrial appendage (LAA). Blue circle: location of the LAA in a transverse cut of the cardiac computed tomography image.

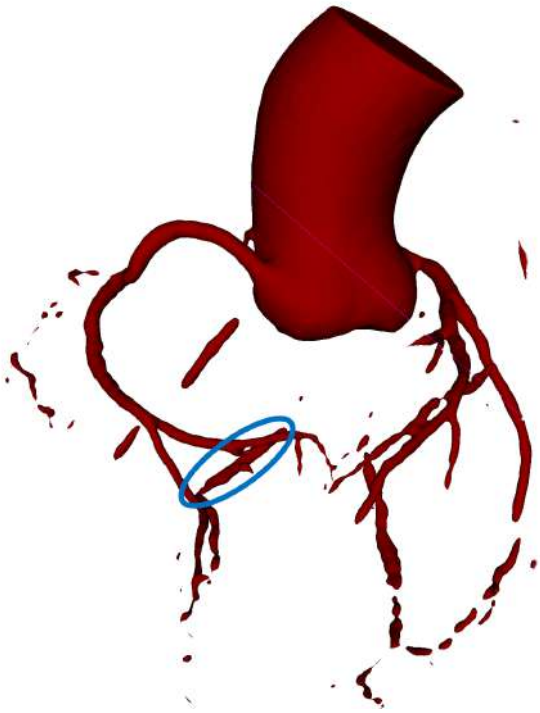


Figure A5: Example of a segmented vein within the pericardium. Blue circle: location of that vein in the 3D reconstruction of the segmentation.

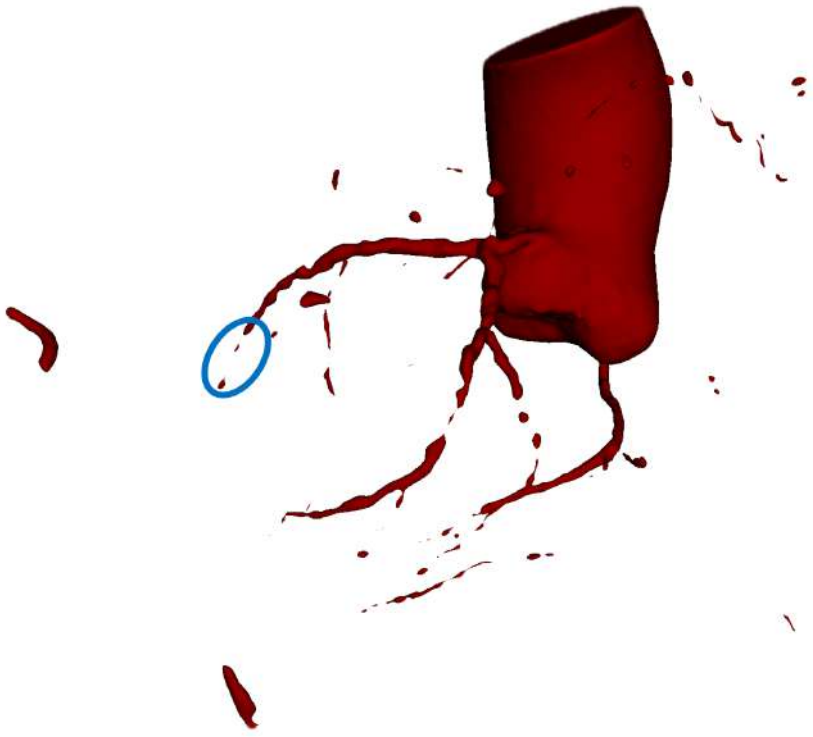


Figure A6: Example in which a distal segment was incorrectly segmented. Blue circle: location of the incorrect segmentation.

Appendix B

Second Appendix

B.1 Workflow proof-of-concept

B.1.1 Segmentation and mesh generation

The first step of the workflow was to use the previously trained segmentation models to achieve segmentation of the coronary artery. For this, a DICOM image of a CCT was selected and then converted to NIfTI to match the input requirements of the nnU-Net framework. Then, it was introduced to the coronary segmentation tool, firstly, and to the pericardium segmentation one, and the inference was therefore run. The result of such computation were two NIfTI binary masks belonging to the coronary artery, on the one hand, and to the pericardial sac, on the other.

With these binary masks, to carry out the simulations, the next step was to obtain the final coronary volume. For this, the process outlined in Section 2.3.3 (Figure 16 was followed, yielding a binary mask corresponding to that volume. Subsequently, this binary mask was converted into an STL file using Slicer.

B.1.2 Mesh post-processing

After creating the STL file, it was not on the correct scale because the data was in millimeters, and the used software, Simvascular, used centimeters as a length

unit. This problem was solved by scaling the mesh by a factor of 0.1 using the filter *Transform*[1] in Paraview.

The scaled STL file was then opened in Meshmixer to perform the post-processing of the mesh. It consisted first of segmentation error corrections, followed by removing the small vessels (no diagnostic utility in this case) and being the final result validated by an expert cardiologist. Then, the number of triangles was reduced using the *Reduction* tool, selecting 40,000 triangles in the option *triangle budget*. Then, each coronary artery was cut using the *plane cut* option, thus ensuring that the outflow was perpendicular to a plane. Moreover, both the ascending aorta and the aortic root were also cut using the same tool. Figure B1 displays the mesh after the cuts.

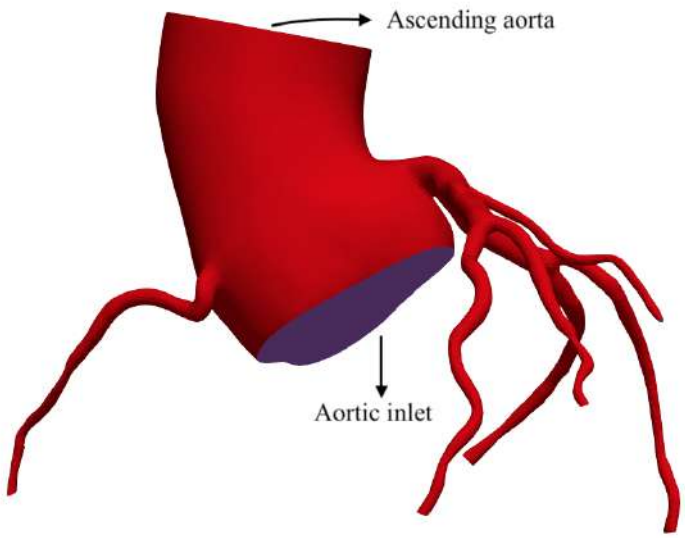


Figure B1: Mesh obtained after the edition of the distal segments of the coronary arteries. In red, the face corresponding to the vessel wall. In purple, the faces corresponding to the coronary and the aorta inlets and the aorta outlet.

Finally, the self-intersecting faces and the elements of the mesh that presented some problem were corrected by using the software Materialise 3-Matic[2]. At that point, the mesh was ready to be introduced to Simvascular to start with the steps of the CFD simulation.

[1]https://docs.paraview.org/en/latest/UsersGuide/filteringData.html?#transform
[2]https://www.materialise.com/en/software/3-matic

B.1.3 Computational Fluid Dynamics simulation

Boundary conditions computation

Although the mesh was ready, the boundary conditions still needed to be computed. For this, the cardiac output of the patient should be calculated based on multi-modality imaging data. Firstly, the area of the right ventricular outflow tract (RVOT) has been measured in the CCT image, resulting in an area of $600.68mm^2$ ($6.0068cm^2$), as displayed in Figure B2.

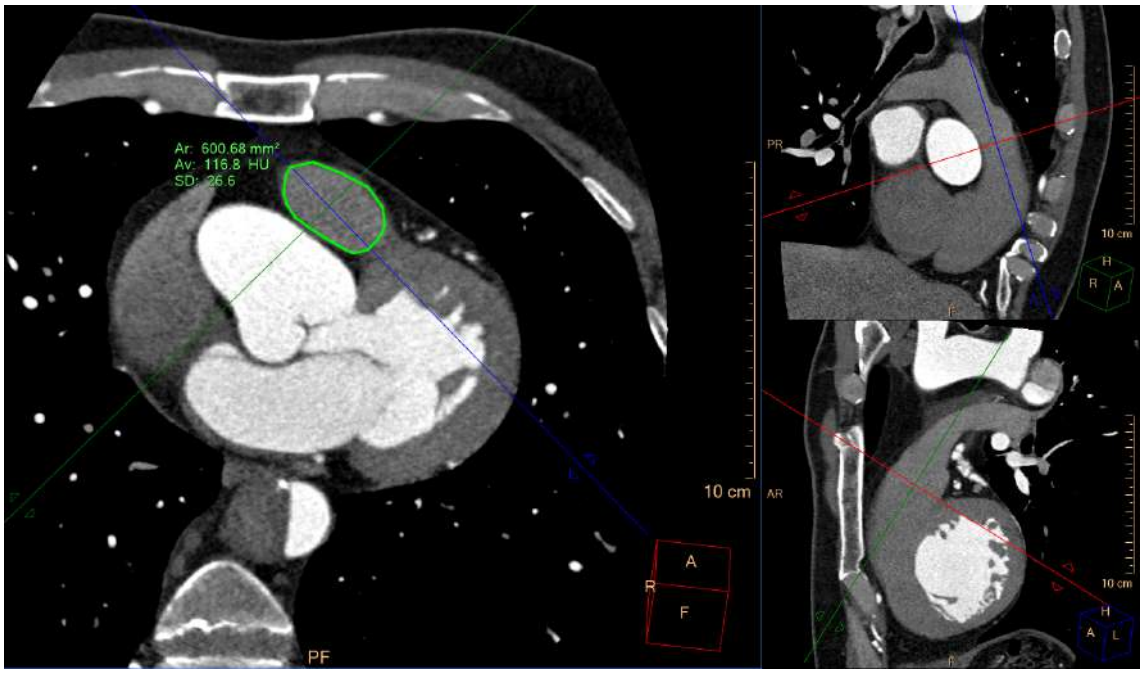


Figure B2: CCT image highlighting the contour of the measured RVOT area (green contour). CCT: cardiac computed tomography, RVOT: right ventricular outflow tract

Then, the VTI (velocity integral over time) was calculated on Pulsed-Wave Doppler echocardiogram images. Figure B3 shows the flow velocity spectrum of the patient over two beats, which was used to calculate the VTI by averaging two beats. In this patient, the VTI was equal to $19.3cm/beat$.

After computing these two elements, the stroke volume (SV) could be calculated. Since it is the result of multiplying the centimeters of blood that have flowed through the RVOT by the area of the RVOT[3], the final result was $SV = area * VTI =$

[3]The SV is defined as the product of the length of the blood column that has flowed through

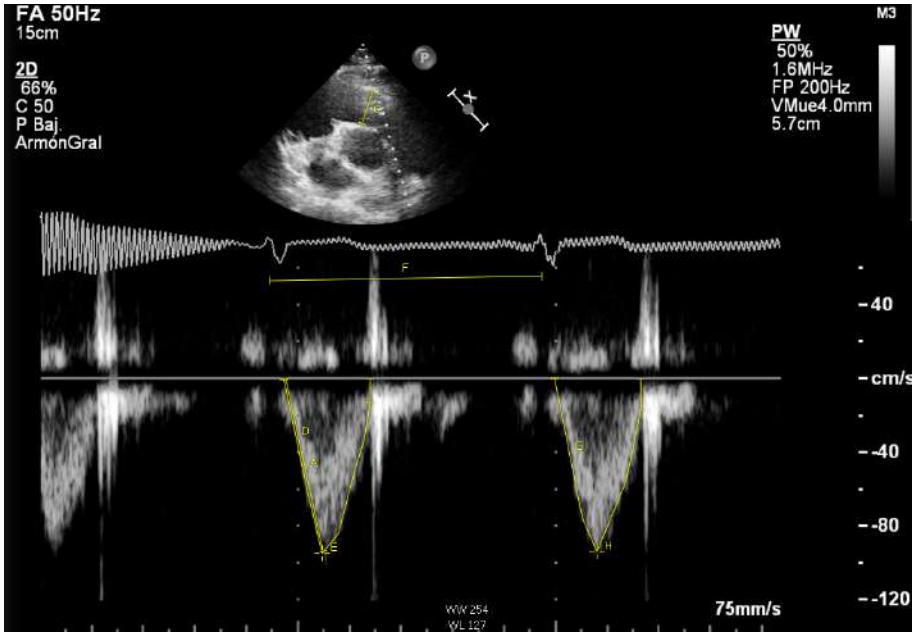


Figure B3: Pulsed-Wave Doppler mode used to compute the velocity integral over time.

$6.0068cm^2 * 19.3/beat = 115.93cm^3/beat$. Hence, by multiplying the SV by the heart rate (which in this case was 60 bpm), the cardiac output could be obtained, which was equal to $115.93cm^3/beat * 60bpm$, i.e., $6954cm^3/min$.

Simulation

After obtaining the data for the boundary conditions, the mesh was introduced into Simvascular, where firstly, the software allowed to select each face of the mesh and determine whether it was a cap or a wall. Then, after establishing each of these conditions to each face, the software discretized the domain using 3D Delaunay Triangulation using the tetrahedral mesh generator "TetGen" [60]. After the creation of the volumetric mesh using this approach, its number of tetrahedra was reduced to 825,000 by manually adapting the resolution of the mesh depending on the finer and coarser sections of the model, as displayed in Figure B4. This way, the model was more computationally efficient than the originally created one, which had more than 150,000,000 elements.

the left ventricle outflow tract by its cross-sectional area, but since blood circulation is a closed system, the result is the same as using the RVOT.

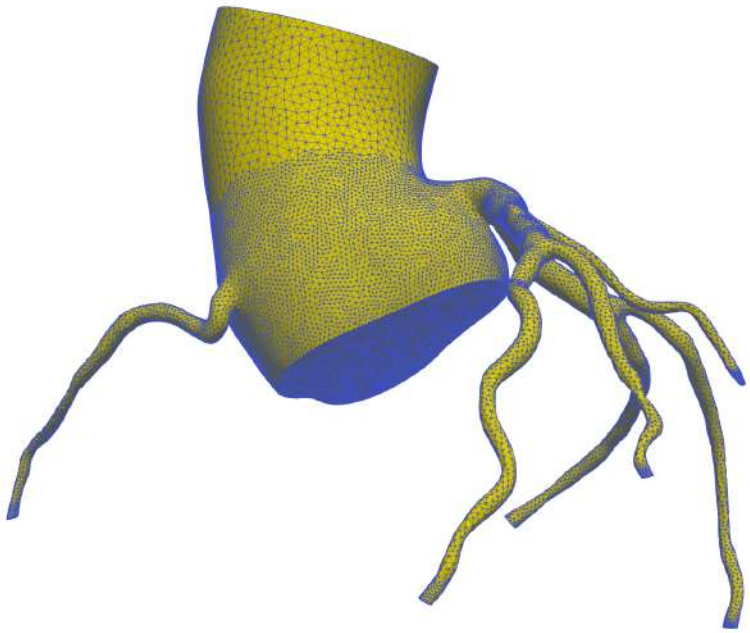


Figure B4: Volumetric mesh displaying the external faces of the 3D discretization. Yellow: faces of the 3D elements. Blue: edges of the 3D elements. The distal segment of the aorta is composed by larger tetrahedra, while the distal segments of the coronary artery present finer elements.

The CFD model was based on that of [61]; thus, the particular boundary conditions for the inflow and outflow aorta and the coronary arteries were determined as in that work, but considering the specific characteristics of the patient. In this sense, the aortic inflow waveform of the model is shown in Figure B5.

For the aortic outflow, a Windkessel RCR boundary condition was implemented [62] as shown in Figure B6a. This way, The sum of the resistances corresponds to the total vascular resistance, while the capacitance represents the elastic capability of the artery. Concerning the coronary arteries, the boundary conditions were implemented according to [61], by making use of a lumped parameter coronary vascular model, as shown in Figure B6b. The walls were simulated subject to a no-slip and no-penetration boundary condition on the flow. It was also necessary to specify the intra-myocardial pressure. For this purpose, a file in which left ventricular pressure was determined from a previous simulation of the cardiac model was used. This file was subsequently edited to achieve consistent results by multiplying all pressures by 1.5 for left coronary outflows and by 0.5 for right coronary outflows.

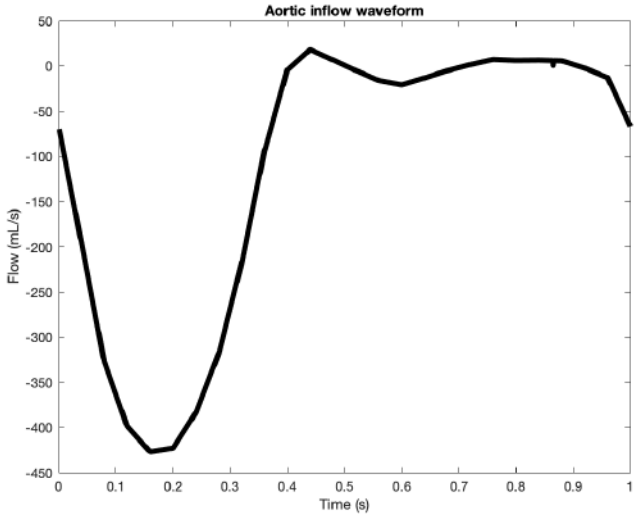
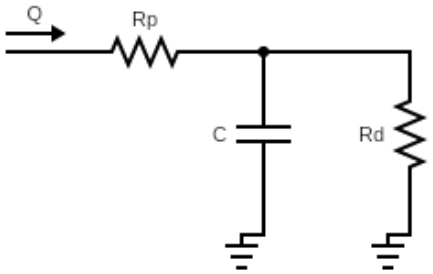
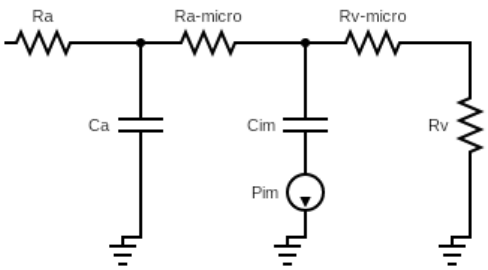


Figure B5: Aortic inflow waveform.



(a) RCR windkessel.



(b) Coronary outlet lumped parameter network (LPN).

Figure B6: Circuits depicting the boundary conditions for the aortic and the coronary outflows.

Finally, the parameters for the solver were set. Two hundred time-steps, each of them with a size of 0.03, were selected. By selecting these arguments, the temporal resolution would be sufficient without significantly increasing the computational expense. With these settings, the simulation was computed. After that, the variables that were exported were pressure, velocity, and WSS.

B.1.4 Visualization of the results

Computer-based visualization

The next step was to get some visualizations that would allow these descriptors to be analyzed in an easy and visually appealing way with the variables exported.

The first visualization consisted of presenting the WSS obtained on the vessel wall of the entire model. In addition, other dynamic visualizations were performed. For example, the flow within the model was also represented by flow lines describing the direction of blood and its velocity.

3D printing

Apart from displaying the design on a (two-dimensional) computer screen, another option is to 3D-print it. Thanks to state-of-the-art technologies, it is possible to achieve colored 3D printed models. In this work, the 3D model was hollowed, displaying the WSS values on its wall, and, additionally, one of the transverse cuts of the aorta was added, displaying the values of flow velocity that pass through it on a determined time step.

The 3D model combining that information was created in Paraview and was subsequently edited in Meshmixer to ensure its stability by adding a supporting structure. The model was finally printed with HP Jet Fusion 580 Color 3D Printer.