



POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Deep Learning based Coronary Artery Centerline tracking aimed at Fractional Flow Reserve Prediction from CCTA images

**TESI DI LAUREA MAGISTRALE IN
AUTOMATION AND CONTROL ENGINEERING -
INGEGNERIA DEL CONTROLLO E DELL'AUTOMAZIONE**

Author: **Matteo Leccardi**

Student ID: 945122

Advisor: Prof. Marco Marcon

Co-advisors: Prof. Gian Franco Gensini,
Ph.D. Marco Brando Mario Paracchini
Academic Year: 2021-22

Abstract

Coronary artery disease (CAD) is characterized by an inadequate supply of oxygen-rich blood to the myocardium because of narrowing or blocking of a coronary artery by atherosclerotic plaques. In extreme cases, the effect may be a myocardial infarction. CAD accounted for approximately 12.6% of deaths in the US in 2018, causing 360'900 deaths. Fractional Flow Reserve (FFR) assessment is a relatively novel invasive procedure for CAD evaluation. The severity and anatomical extent of CAD can be assessed non-invasively by Coronary Computed Tomography Angiography (CCTA). Lately, many approaches are being proposed to fully-automatically estimate the FFR index directly from CCTA images (FFR_{CT}). In this work, state-of-the-art FFR_{CT} estimation approaches are exhaustively analysed in each and every step involved. All rely on one mandatory preliminary step, which is automated coronary artery centerlines estimation. Recently, Wolternik et al. proposed a CNN-based tracking scheme which iteratively tracks the location of the centerline, obtaining state-of-the-art performance as a deep learning-based centerline extraction algorithm. Their tracker required multiple seeds per arterial branch to successfully extract the related centerlines, and fails when too few seeds points are initialised. Three new CNN architectures are proposed, as well as a new tracker capable of reconstructing the full arterial branch (left or right) starting from just one seed point planted inside a coronary artery. One of the CNN proposed was able to outperform the state of the art method in both the ability to retain information it was trained on and tracking accuracy. Qualitative artery tree reconstruction results show also that both the accuracy and tracking capability of the proposed tracker surpass the considered baseline, even in prohibitive testing conditions. The advantages of adopting a highly modular tracker through object-oriented programming are discussed. Moreover a post-processing step based on graph-algorithms is introduced to classify and reconstruct a topologically consistent map of the left and right coronary artery trees. Two standard test suites are built to test and compare the proposed CNNs and trackers.

Keywords: coronary artery centerline tracking, coronary computed tomography angiography, deep convolutional neural networks, Fractional Flow Reserve, coronary artery disease.

Abstract in lingua italiana

La coronaropatia è caratterizzata da un apporto inadeguato di sangue ossigenato al miocardio a causa del restringimento o dell'ostruzione di una arteria coronarica da parte di placche aterosclerotiche. In casi estremi, l'effetto può risultare in un infarto. La coronaropatia è stata la causa del 12,8% dei decessi negli Stati Uniti nel 2018, provocando 360'900 morti. La valutazione dell'indice Fractional Flow Reserve (FFR) è una procedura invasiva relativamente nuova per la valutazione della coronaropatia. La gravità e l'estensione anatomica della coronaropatia può essere valutata in maniera non invasiva da un'angiografia con tomografia computerizzata (Coronary Computed Tomography Angiography - CCTA). Recentemente sono stati proposti diversi approcci per valutare automaticamente l'indice FFR direttamente dalle immagini ottenute tramite CCTA (FFR_{CT}). In questo documento, gli approcci allo stato dell'arte della valutazione dell' FFR_{CT} sono analizzati esaustivamente in ogni fase coinvolta. Ognuno di essi si basa su un primo stadio obbligatorio, ovvero la valutazione automatizzata delle linee centrali del lume delle coronarie. Recentemente, Wolternik et al. hanno proposto un sistema di tracciamento basato su reti neurali convolutive (CNN) che traccia iterativamente la locazione delle linee centrali, ottenendo prestazioni allo stato dell'arte per quanto riguarda metodi di estrazione basati sul deep learning. Il loro tracciatore richiede diversi punti di origine per ramo arterioso al fine di estrarre con successo le relative linee centrali, e fallisce se non vengono inizializzati sufficienti punti di origine. In questo documento sono proposte tre nuove architetture CNN che superano in prestazioni quella proposta nel lavoro di Wolternik, oltre che ad un nuovo tracciatore capace di ricostruire un intero ramo arterioso (sinistro o destro) iniziando da un solo punto di origine e che traccia le linee centrali in media con maggiore accuratezza. Sono discussi i vantaggi di adottare un tracciatore altamente modulare attraverso una programmazione ad oggetti. Viene inoltre introdotta una fase di post-elaborazione basata sui grafi per classificare e ricostruire una mappa topologicamente coerente degli alberi coronarici sinistro e destro. Due suite di test sono state costruite per confrontare le CNN e i tracciatori proposti.

Parole chiave: tracciamento automatico arterie coronarie, angiografia tomografia computerizzata, reti neurali, Fractional Flow Reserve, cardiopatia coronarica.

Contents

Abstract	i
Abstract in lingua italiana	iii
Contents	v
1 Introduction	1
1.1 Fractional Flow Reserve index (FFR)	2
1.2 Current clinical problems and possible solutions	3
1.3 Structure of the document	4
2 FFR estimation methods from CCTA	7
2.1 Methods based on fluid dynamic simulations	8
2.2 Data-driven black-box methods	12
2.2.1 Lumen segmentation and hybrid approaches	13
2.2.2 Standalone coronary artery tree centerlines extraction	15
2.2.3 Lumen segmentation subsequent to centerlines extraction	20
2.2.4 FFR estimation	23
2.3 Main matter declaration and context	27
3 Methods: the 3D-CNN Tracker and its descendants	31
3.1 Neural networks, the core of the tracker	33
3.1.1 Wolternik's CNN	37
3.1.2 First proposed network: input-layer-split	39
3.1.3 Second proposed network: midsplit-4	40
3.1.4 Third proposed network: midsplit-4-FD	42
3.2 Dataset and training strategies	42
3.2.1 Training, validation, cross-validation and testing	43
3.2.2 Wolternik's training strategy and its re-implementation	46

3.2.3	Further training strategies	48
3.3	Centerline trackers	49
3.3.1	The basic Tracker class	49
3.3.2	Wolternik's tracker	50
3.3.3	Seedspawner tracker	51
3.4	Post-processing centerlines graphs	53
4	Test suite and results	59
4.1	Local test suite for CNN assessment	59
4.1.1	Local tests results discussion	62
4.2	Global test suite for tracker assessment	67
4.3	Results discussion	70
5	Conclusions and future developments	77
Bibliography		81
A	On coronary arteries and coronary artery disease	97
A.1	The heart	98
A.2	Coronary vasculature system	98
A.3	Coronary arteries disease and atherosclerosis	100
A.4	Common revascularisation procedures	103
B	On CCTA images	107
C	On supervised machine learning	111
D	On coronary artery tree extraction from other imaging sources	115
E	On the Rotterdam Coronary Artery Algorithm Evaluation Framework	117
E.1	Training set, test set and reference standard	118
E.2	Standard evaluation framework	118
List of Figures		123
List of Tables		129

1 | Introduction

Obstructive coronary artery disease is the most common type of cardiovascular disease [8]. It develops when atherosclerotic plaque builds up in the inner wall of the coronary arteries¹, narrowing the artery lumen. This is defined as coronary stenosis (from Ancient Greek στενός, "narrow"), which can potentially limit blood supply to the myocardium and could lead to ischemia and irreversible damage. Appendix A proposes a focused introduction on the heart and its vasculature, as well as coronary artery disease description, causes, and statistics from the American Heart Association [89].

Only stenoses which significantly limit blood flow need to be invasively treated to reduce disease morbidity. Contrarily, invasively treating a functionally non-significant stenosis may lead to harmful consequences and exposes the patient to an unnecessary factor of risk. Therefore, it is crucial to assess the functional significance of a coronary stenosis to guide treatment. To this end, severity assessment can be performed in many ways, however some are found to be more efficient than others.

As Edoardo Verna and Tommaso Gori pointed out in *Atlas of FFR-Guided Percutaneous Coronary Interventions* [28], studies and daily clinical experience consistently show that the accuracy of coronary angiography, which in the clinical routine continues to be used as the standard method for quantifying the severity of coronary stenoses, is very modest. Moreover, although intravascular ultrasound and optical coherence tomography do provide valuable information on coronary anatomy, they fail to give any useful insight on coronary hemodynamics. Therefore, these methods are deemed unfit [28] to help in the decision whether a stenosis should be treated or not.

In contrast, measures of intracoronary pressure provide information on the hemodynamic impact of single and multiple atherosclerotic lesions. The most common devices currently available on the market are based on a microscopic pressure sensor placed on the tip of a wire ([17] and figure 1.1) and allow numerical quantification of the impact of a specific stenosis and of the benefit that can be achieved with invasive revascularization procedures.

¹Coronary arteries are the arterial blood vessels which transport oxygenated blood to the heart muscle, and they wrap around the outside of the heart. Small branches dive into the heart muscle to bring it blood. The matter is further discussed in Appendix A.

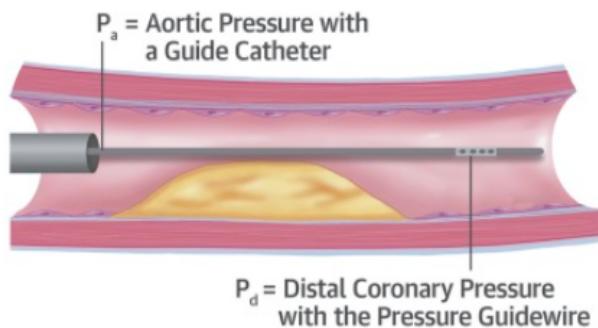


Figure 1.1: Illustration of a catheter-guidewire pressure measuring system. The catheter enters from the aorta and the guidewire slides towards distal regions of the coronary vessel. Source: Kobayashi et al., 2015 [44].

Although a few different metrics based on intracoronary pressure have been proposed and used throughout the years, the most used one is based on the assessment of P_a (aortic pressure, or pressure proximal to a stenosis) and P_d (pressure distal to -after- the stenosis), as illustrated in figure 1.1.

1.1. Fractional Flow Reserve index (FFR)

Fractional Flow Reserve (FFR) is defined under (maximal) hyperemia² as the ratio between maximum blood flow distal to a stenotic lesion P_d and normal maximum flow in the same vessel P_a , and can be computed from the aforementioned pressure measures: $FFR = P_d/P_a$. This results in an absolute number comprised theoretically between 0 (full occlusion without collaterals) and 1 (no functional effect of the stenosis).

For instance, an FFR of 0.8 corresponds to a 20% drop in blood pressure and, therefore, reduction in relative blood flow across the vascular segment under study and reduced perfusion capacity in the portion of myocardium served by the vessel and its branches. This specific value (FFR=0.8) is normally recognized as the threshold under which a stenosis or lesion is classified as functional significant [2]. The FFR index is currently considered the golden standard to reliably assess the hemodynamic relevance of a coronary stenosis, as hundreds of clinical studies and publications have gathered solid evidence in favour of improved clinical outcomes when coronary revascularization is indeed guided by FFR measurement instead of coronary angiography [2].

²Hyperemia is a state in which the coronary epicardial, arteriolar, and microcirculation are fully dilated, resulting in the distal coronary bed's resistance being minimal and constant; this condition can be obtained with drugs such as adenosine or papaverine. [76] presents the basic definitions and concepts of FFR, a focused understanding of the need for hyperemia during assessment and key clinical studies supporting its use. Other indices are being investigated that do not require the maximal hyperemia condition [44].

A specific study [71] showed a weak correlation between angiography, expressed as a percentage of stenosis diameter, and FFR (correlation index was only 33%). This disagreement between stenosis diameter and physiology has already been documented in several other studies, showing disagreement rates between FFR and angiography of 36% [87] and 39% [65]. One large clinical registry demonstrated that performing FFR changed further management in 43% of patients compared to the purely visual assessment of coronary artery stenoses by invasive angiography [7].

In the FAME (Fractional Flow Reserve Versus Angiography for Multivessel Evaluation) trial of 1005 patients with multivessel coronary artery disease, FFR-guided revascularization for lesions with $\text{FFR} \leq 0.80$ was associated with a 28% lower rate of major adverse cardiac events compared with an angiography-guided strategy [24, 86]. The results from FAME are in accordance with the 5-year follow-up of individuals from the DEFER study [67]. Even when non-invasive proof of ischaemia is available, FFR measurements often change clinical judgement regarding the need to revascularize a given coronary artery stenosis. Later studies have gained evidence of interesting cost-effectiveness [70].

The aforementioned, together with previous studies, led in 2010 to the official endorsement by the European Society of Cardiology in the myocardial revascularization guidelines, granting class IA use of FFR in patients with stable coronary disease in the absence of objective evidence of myocardial ischemia at the non-invasive functional evaluation [28].

1.2. Current clinical problems and possible solutions

Clinical adoption of the FFR index has since increased, but slowly, and it is still scattered in some geographic areas [28]. Despite clear evidence and guidance, many interventional cardiologists continue to rely only on the visual assessment of stenosis severity alone rather than performing invasive FFR analysis in the absence of objective evidence of myocardial ischemia [88].

Potential reasons include the logistical effort of performing FFR, concerns regarding potential complications and uncertainty about optimal performance and interpretation of FFR measurements, particularly in complex situations, such as multi-vessel disease, left main stenoses, serial stenoses or in patients with aorto-coronary bypass grafts. In addition, while performing FFR is not technically difficult *per se*, several relevant procedural aspects must be considered to avoid incorrect measurements or misinterpretation of results [2].

Difficulties in logistics and diffused sentiment of mistrust towards the invasive procedure led researchers to intensively investigate non-invasive determination of the functional sig-

nificance of stenoses based on Coronary Computed Tomography Angiography (CCTA) images, in an attempt to reduce the number of unnecessary invasive procedures. To obtain such information of the coronary artery of a suspicious patient, cardiac CT angiography (CCTA) is widely used because it is non-invasive, it has high sensitivity [47] and the necessary machinery, operators and protocols are widespread and quite common in any major medical facility.

Several studies [27, 62] have demonstrated that CCTA derived metrics such as the luminal dimensions, plaque burden and the composition of the plaque can provide useful prognostic information [3], while Papadopoulou et al. [63] demonstrated that CCTA has a value in assessing changes in the composition of the plaque in patients admitted with an acute coronary event. An accurate 3D reconstruction of the main coronary arteries can provide models that allow comprehensive visualization of the vessel geometry and assessment of the distribution of different plaque types in space; moreover, the 3D reconstruction also allows the estimation of various functional indices, such as the FFR, in a non-invasive way. The reported early studies [61, 100] were able to assess that FFR_{CT} (acronym standing for "FFR computed from CCTA images") provides satisfactory diagnostic accuracy and discrimination for the diagnosis of hemodynamically significant coronary artery disease with invasive FFR as the reference standard. These studies show how CCTA can be reliably used to perform FFR_{CT} measures to estimate the FFR index in the coronary artery tree of a patient, reducing the risks related to an invasive pressure measurement procedure and to the induction of the hyperemia condition.

Before analysing the two main categories of approaches adopted in the literature to estimate FFR values from CCTA images³, a short digression on the images themselves and how they are acquired can be useful. A panoramic view on the most common medical imaging techniques as well as some interesting details on CCTA are reported in Appendix B.

1.3. Structure of the document

In chapter 2 the state of the art regarding the full pipeline of FFR_{CT} estimation is reported and analysed starting from computational fluid dynamic approaches. Deep learning approaches to execute each step of the FFR_{CT} estimation pipeline are described; in particular coronary artery centerline extraction, vessel lumen segmentation, and subsequent

³*Cardiac/Coronary Computed Tomography Angiographies* are not the only type of imaging technique in the literature that has been used to achieve coronary artery tree 3D reconstruction and subsequent FFR estimation. As briefly discussed in Appendix D, some advanced techniques use classic angiographic images, biplane orthogonal angiography, or also CCTA images combined with other invasive or semi-invasive methods.

FFR_{CT} estimation methods are discussed.

In chapter 3 three new convolutional neural networks are described and compared to a baseline network. A new tracker structure is proposed as well, and the benefits of implementing a modular tracker are discussed. A post processing step to automatically create a graph-based oriented map of the coronary artery tree extracted by the tracker is introduced.

In chapter 4, two test suites for the evaluation and comparison of both the CNNs and the centerline trackers against the ones proposed by Wolternik et al. [99] are discussed, and the results commented.

Finally, in chapter 5 the conclusions drawn upon the results of the performed tests are discussed, and some ideas are put forward for future development in the field explored in this document.

In the next chapter, the most relevant state-of-the-art methods regarding the entire automatic FFR_{CT} extraction pipeline are classified into two main classes. Each class is analysed, and common critical issues discussed, eventually leading to the in depth analysis of a specific subclass of very promising, machine-learning based methods.

2 | FFR estimation methods from CCTA

Several automatic methods for the determination of the functional significance of coronary artery stenosis from CCTA images have been proposed throughout the last two decades, and most of them rely on the estimation of the FFR index (namely, FFR_{CT}). Two main categories can be identified while classifying possible approaches to automatically estimate the FFR_{CT} [111].

The first category comprises those methods that simulate and analyse the physics of blood flowing in the coronary arteries. While most methods rely purely on computational fluid dynamics (CFD), there are examples in the literature of machine learning techniques used in aid to fluid simulation environments.

The second category contains all those methods that rely solely on geometrical characteristics automatically extracted from the myocardium and coronary arteries in CCTA scans without involving any simulation of physical phenomenon, predominantly through machine learning techniques.

While the FFR_{CT} estimation procedure can be divided into the two aforementioned categories, the geometric analysis needed to extract the vessels geometry (the line passing through the center of the lumen of the vessel, and the lumen surface) is a prerequisite step to any FFR_{CT} estimation approach. This prerequisite step, which is the main matter of interest of this thesis, is exhaustively discussed in sections 2.2.1, 2.2.2 and 2.2.3.

The discussion now proceeds in section 2.1 by first introducing the most intuitive and common FFR_{CT} estimation approach: blood flow analysis by means of fluid dynamics simulation. Some critical issues in the reported methods are considered in the discussion. The discussion will then proceed in section 2.2 with the less intuitive, but nonetheless promising approaches involving machine learning techniques, illustrating how machine-learning has been and is being successfully used to extract the coronary arteries center-lines, segment the lumen volume and lumen internal wall, and estimate FFR_{CT} in many different ways.

2.1. Methods based on fluid dynamic simulations

Methods that simulate and analyse the blood flow in the coronary arteries in CCTA images estimate FFR_{CT} values along the entirety of the considered coronary arteries, which can be used to determine the functional significance of coronary artery stenosis and the precise location of lesions.

The computation of FFR_{CT} requires the construction of an anatomic model of the coronary arteries: a mathematical model of coronary physiology to derive boundary conditions representing cardiac output, aortic pressure and micro-circulatory resistance, and a numerical solution of the laws of physics governing fluid dynamics. This combination of anatomy, physiology, and fluid dynamics enables the computation of coronary flow and pressure. Specifically, coronary flow and pressure can be computed by solving the governing equations of fluid dynamics, which relate to conservation of mass and balance of momentum, and which have been known in their current forms as the Navier-Stokes equations for more than 150 years:

$$\frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot (\rho \vec{u}) = 0 \quad (2.1)$$

$$\frac{\partial(\rho \vec{u})}{\partial t} + \vec{\nabla} \cdot [\rho \overline{u \otimes u}] = -\vec{\nabla} p + \vec{\nabla} \cdot \vec{\tau} + \rho \vec{f} \quad (2.2)$$

$$\frac{\partial(\rho e)}{\partial t} + \vec{\nabla} \cdot ((\rho e + p) \vec{u}) = \vec{\nabla} \cdot (\vec{\tau} \cdot \vec{u}) + \rho \vec{f} \cdot \vec{u} + \vec{\nabla} \cdot (\vec{q}) + r \quad (2.3)$$

(Here \otimes denotes the tensorial product, forming a tensor from the constituent vectors. A double bar denotes a tensor.)

These equations are solved for the unknown pressure p , which varies with respect to position and time, and for the three components of blood velocity \vec{u} , each of which are functions of position and time. The physical properties of blood, the fluid density and the fluid viscosity, are known when solving these equations: although blood exhibits complex rheological properties, it can be approximated as a Newtonian fluid with a constant viscosity in large arteries.

Notable features of the equations of fluid dynamics are their universality for describing phenomena ranging from airflow over a jet plane wing to water flow in a river to blood flow in arteries, as well as their ability to capture complex phenomena such as turbulent flow (situation for which, as of today, there is no closed-form solution for a 3-dimensional description, nor its existence has been proved).

However, the governing equations of blood flow can only be solved analytically under special circumstances (e.g., steady or pulsatile flow in an idealized circular cylindrical geometry). For realistic patient-specific models of the human coronary arteries, a numerical method must instead be used to approximate the governing equations and to obtain a solution for velocity and pressure at a finite number of points [81]. This requires solving millions of nonlinear partial differential equations simultaneously and repeating this process for thousands of time steps in a single cardiac cycle. Numerical methods for solving fluid dynamics problems are known, in fact, as computational fluid dynamics (CFD) methods. Because of the complexity of the numerical solution of this class of problems, computation times are usually high, and the entire process is highly resource-demanding. It also has to be noted that the governing equations alone are insufficient to solve blood flow problems: a domain of interest¹ must be defined, and boundary conditions² must be specified.

Despite the challenging problem definition and the resource-consuming numerical approximate resolution, researchers proposed a number of different approaches for the computation of the FFR_{CT} by using fluid dynamic simulations nonetheless, because fluid dynamic simulations are a well-known framework and this method is conceptually intuitive and thus retains a certain appeal for potential clients.

Taylor et al. [82] were the first, in 2013, to propose a non-invasive, flow-based approach to compute the FFR_{CT} , which was later validated in multiple clinical studies [19, 51, 61]. Computation of FFR_{CT} required coupling lumped parameter models³ of the heart, systemic circulation, and coronary micro-circulation to a patient-specific model of the aortic root and epicardial coronary arteries extracted from CCTA data. In the case of CFD methods, very accurate lumen segmentation (section 2.2.3) is fundamental to obtain correct results. The finite element method used to solve the fluid dynamic problem built a mesh generated from the geometric model with millions of vertices and elements, and the numerical solution for velocity and pressure was computed on a parallel supercomputer. Also Itu et al. [35] presented in the same years a method to estimate FFR in the coronary artery tree in CCTA images by simulating blood flow. This method used a parametric lumped heart model as well, while modelling the patient-specific hemodynamics in both

¹Specific to modelling blood flow in arteries, the domain of interest is where the blood is flowing (the lumen), and some of the relevant boundaries are the lateral surface, the inlet boundary (the aortic root), and the outlet boundaries of the ascending aorta and the coronary arteries.

²Boundary conditions are mathematical relationships between the variables of interest (like the flow - pressure relationship) defined on the boundaries of the mathematical model.

³The term *lumped* comes from electrical engineering practices, and refers to lumped-parameter analysis, opposed to distributed-parameter analysis. In a lumped-parameter model spatial variations of parameters are ignored and the system is described by discrete entities - each associated to one or more parameters - that approximate the behaviour of the distributed system under certain assumptions.

healthy and diseased coronary arteries thus considering a broader spectrum of interactions, such as between different branches.

Nickisch et al. [60] presented in 2015 a method to determine FFR_{CT} values along a whole coronary artery by simulating blood flow and pressure along the coronary arteries using an electrical parametric lumped model.

Moreover, Ituet et al. presented in 2016 a machine learning based model for estimating FFR along the coronary artery[36]. The model is trained on a large number of synthetically generated coronary anatomies, where the target values are computed using a blood flow-based model [35]. This method is further evaluated in [16].

All methods reported above exploit the simplicity of reduced order lumped parameter models for the inlet and outlets of the coronary arterial tree to approximate the patient-specific boundary conditions, and suffer from the inability to specify the associated parameters in a patient-specific manner.

The proof-of-principle work presented by Moore et al. [58] point out that when using X-Ray angiograms as the basis for creating the geometrical model of the coronary artery tree, there is some indirect functional information available through observing the motion of the radio-opaque contrast agent. They propose to exploit these information for tuning the arterial resistances in the lumped parameters model specifically for each patient, based on the simulation of the contrast release and the minimisation of the misfit defined in terms of simulated and observed arrival times of the contrast at multiple points within a coronary tree.

Currently, HeartFlow® is the leading lucrative company for on-demand, non-invasive personalized cardiac test, and it bases its analysis on a CFD approach. Figure 2.1 illustrates their core coronary arteries analysis pipeline. Their use of artificial intelligence mostly regards heart and vessels segmentation and the final interpretation of data obtained from fluid-dynamic simulations. This is the core business of a company based in the United States of America, where the healthcare system is notoriously a very expensive and lucrative sector, especially for advanced analysis such as this one. Such a system allows the company to afford the supercomputers needed to run the very resource-demanding simulations in a substantially shorter time than would be possible for any other entity - especially if publicly funded in a welfare-state country. Currently, thanks to their business model and location, HeartFlow® attracts all the best minds in the field of matter and has the best technology and advanced approaches available (such as [64], discussed in details at the end of section 2.2.4).

Although techniques based of CFD achieved high accuracy, they are remarkably dependent on the accuracy of coronary artery lumen segmentation [84]. To achieve accurate

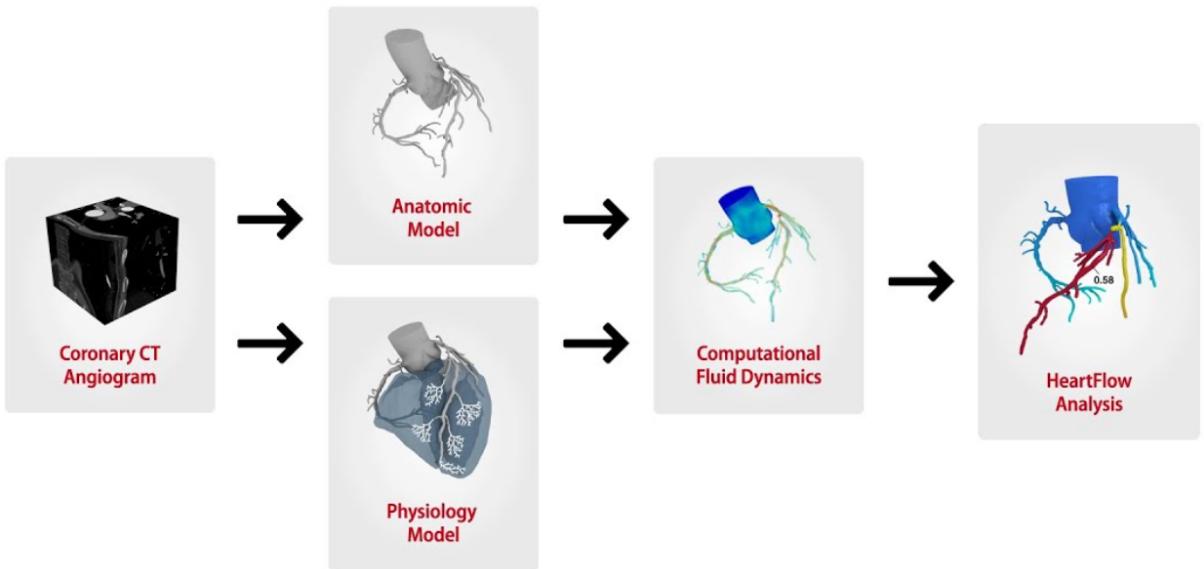


Figure 2.1: HeartFlow® FFR_{CT} estimation pipeline based on CFD. Source: heartflow.com.

segmentation, manual annotation of the coronary artery lumen is needed, which is a very time consuming and complex task: commercially available automatic software tools typically required substantial manual interaction and correction, especially in CCTA scans with excessive atherosclerotic calcification or imaging artefacts due to stents and cardiac motion [42].

Even when the latest state-of-the-art lumen segmentation algorithms are used (overviewed in section 2.2), it still has to be considered that these methods rely on a complete fluid dynamic simulation for each patient, which is *per se* a computationally and time intensive task [43, 61] and requires dedicated hardware in order to be completed in a reasonable time and with satisfactory resolution and accuracy.

These methods met great favour from the medical community despite being very resource-intensive, and they are trusted to yield accurate results. Trust in fluid-dynamic simulations is not only driven by trials and objective proofs, but are also influenced by an important psychological factor: to the end user of these technologies it makes sense to study blood flow and pressure-derived metrics, such as FFR, by using an algorithm that simulates the motion of the fluid, as it is the method that most closely approximated reality.

Until recently, fluid-dynamics methods were the only approaches yielding sufficiently accurate and reliable results to be actually used in clinical practice. More recently, methods that do not model the blood flow in the coronary arteries but only employ characteristics extracted from the arteries and myocardium in CCTA scans have shown to be not

only feasible, but also provide many positive improvements with respect to CFD methods in terms of computational time and used resources, at the expense of a less immediate simplicity in the interpretation of the methods and a burdensome initial data collection phase.

2.2. Data-driven black-box methods

Purely data-driven, black-box methods have recently gained relevance in the estimation of the FFR index from CCTA scans through the FFR_{CT} . Machine learning algorithms provide information in a more objective, reproducible, and rational manner and with improved diagnostic accuracy in comparison to CCTA [83].

There has been an intensive work in both research and medical validation of data-driven methods in the last years, also thanks to a renovated curiosity towards machine learning algorithms and architectures from all branches of science, vastly validated by ground-breaking results in many different fields.

The advantages of using such methods are beginning to outweigh those of methods involving physical, fluid dynamic simulations [43, 61, 96].

Since it is common experiencing a certain initial difficulty in fully appreciating the principles behind automatic data-driven methods which could lead to mistrust towards this category of approaches, a quick primer on common machine learning methodologies is proposed in Appendix C; the appendix also offers an overview of the theoretical advantages and drawbacks of machine learning techniques which is very useful for better framing the main issues and strength of this approach.⁴

Being machine-learning a very general-purpose framework with little constraints in its applications, different paths can be followed in order to achieve FFR estimation from raw CCTA images by using data driven methods. Every paper presents its own original procedure, but some common-ground steps can be extrapolated – some of which are executed through standard image analysis, other through deep learning techniques. In order to analyse the state-of-the-art in the most methodical way, three steps are considered.

As anticipated at the beginning of chapter 2, most methods in the literature divide the FFR estimation procedure in two moments. In the first moment, the geometry of the coronary artery tree is extracted. This moment has the objective of reconstructing the 3D shape of the coronary artery tree by finding the arteries center-lines and, in many cases, performing lumen segmentation⁵. These two steps are closely interlaced: in older

⁴Additional resources can be found in a comprehensive work from Chen et al., 2020 [13].

⁵In digital image processing and computer vision, image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels). Image segmentation is typically used to locate

approaches lumen segmentation is performed straightaway and oftentimes it is a prerequisite to coronary centerlines extraction, while novel deep-learning techniques proceed either by first extracting the coronary artery centerline (section 2.2.2) and then performing segmentation, or by an hybrid, interlaced approach (as discussed later in this chapter in section 2.2.3).

In the second and final moment, the previously reconstructed trees are exploited to perform the actual estimation of the FFR_{CT} , which can be performed in many ways following different approaches.

While this general procedure (centerline extraction \rightarrow lumen segmentation \rightarrow FFR_{CT} estimation) is followed by almost all considered state-of-the-art approaches, some prefer to not explicitly distinguish the geometric analysis moment from the FFR estimation moment, or the FFR estimation moment from the lumen segmentation step. It has to be noted that when deep neural networks are used the geometric features extraction part is indeed performed by the network itself in its first layers – the step is hidden, but it always exists within the procedure.

In order to be coherent with this common decomposition of the general procedure, the state-of-the-art analysis of machine-learning based methods proceeds as follows: some noticeable results treating exclusively the geometric reconstruction of the tree are discussed first, starting from hybrid lumen-centerline extraction approaches; the discussion then proceeds to the standalone centerline extraction task and subsequent lumen segmentation task, eventually discussing some noteworthy FFR estimation techniques in a chronological order.

2.2.1. Lumen segmentation and hybrid approaches

Before diving into the main matter of this chapter, it has to be noticed that in the literature the lumen segmentation and centerline extraction tasks are not exclusively performed starting from CCTA images, but also from other medical imaging sources. Despite it not being the main matter of this document, some noteworthy results are reported in Appendix D for reference, to acknowledge that efforts are being made in solving the same tasks starting from different image data.

Lumen segmentation refers to the activity of extracting and mathematically representing the volume internal to the inner walls of a coronary artery – where the blood flows. The output of this step varies with respect to the proposed approach, and the most common outputs are:

objects and boundaries. Coronary artery centerline extraction and lumen segmentation are the first but crucial steps in the image analysis process as it allows detection of stenosis and high-risk plaques.

- a volume represented by all those voxels (pixels in 3 dimensions) residing inside the arteries;
- a 3D surface, represented by a surface mesh;
- a 3D point cloud, which is however converted into a surface mesh.

Another subdivision has to be considered, which is whether the procedure involves some kind of machine-learning technique (generally this happens in novel approaches) or, more classically, consists just in an algorithmic approach. In this instance, some noteworthy examples of both kinds are discussed.

A third, less relevant subdivision can be considered: some methods perform just lumen segmentation without considering the centerline extraction, while others (the majority) extract the centerline from the segmented lumen.

Kirbas et al. [41] in 2002 and Lesage et al. [48] later in 2009 reviewed various classic vessel lumen segmentation methods based on algorithmic image analysis.

The work from Athanasiou et al. in 2016 presented a methodology for 3D reconstruction of the coronary arteries and plaque morphology using exclusively CCTA volumes by means of classic image analysis techniques [5]. The steps followed by this approach are common to all classic algorithmic image analysis procedures: first, initial raw images are preprocessed in order to enhance valuable features (the coronary artery tree lumen); then, often, a first rough estimation of the lumen and outer vessel wall borders and approximation of the vessel's centerline is performed; after this step, manual intervention is usually needed to perform precise adaptation of plaque parameters to avoid geometric misinterpretations; finally, accurate extraction of the lumen centerline is performed, enabling detection of the lumen wall borders (and calcium plaque region), and 3D surface construction of the internal lumen of the arteries. Specifically, Athanasiou et al. validated their results against a state-of-the-art IVUS plaque characterization method (see Appendix D), and concluded that "the proposed methodology is fast and accurate and thus it is likely in the future to have applications in research and clinical arena", proving that CCTA extracted coronary morphology can be superior with respect to other imaging techniques.

Although this method proves the feasibility of using standalone CCTA images, it (and generally this family of methodologies) often requires substantial human intervention slowing down the process of reconstruction, and it is based on fairly common and classical image processing techniques (such as contrast enhancement filters, image thresholding, algorithmic edge detection, etc.). A non-trivial drawback of classic image analysis approaches is that particular care has to be reserved to the impact of the vessel lumen segmentation threshold, as the setting of this threshold a little lower or higher than the a-priori unknown optimal value can lead to artefacts such as vessel rupture or indiscriminate engorgement

and thus incorrect lumen segmentation, possibly invalidating all the subsequent analysis [15, 90].

Deep learning has become the most widely used approach for cardiac image segmentation in recent years. This remarkable work from Chen et al. [13] in 2020 reviews over 100 cardiac image segmentation papers using deep learning, which covers common imaging modalities including magnetic resonance imaging, computed tomography, and ultrasound and major anatomical structures of interest (ventricles, atria, and vessels). They report that up to then only a small number of works investigate the use of deep learning in the context of coronary artery lumen segmentation and subsequent centerline extractions and tree reconstruction.

Most deep learning based approaches use an end-to-end convolutional neural network (CNN) segmentation scheme to predict dense segmentation probability maps [33, 53, 57, 77]. Merkow et al. [53] introduced deep multi-scale supervision into a 3D U-net architecture, enabling efficient multi-scale feature learning and precise voxel-level predictions.

Shape priors can also be incorporated into the network and has been examined as well [14, 20, 46]. For instance, in 2019 Lee et al. [46] explicitly enforced a roughly tubular shape prior for the vessel segments by introducing a template transformer network, through which a shape template can be deformed via network-based registration to produce an accurate segmentation of the input image, as well as to guarantee topological constraints.

Centerlines extraction and lumen segmentation are related, yet they are sometimes treated separately in the literature. In the following section the most promising state-of-the-art coronary artery centerlines extraction algorithms are introduced.

2.2.2. Standalone coronary artery tree centerlines extraction

Extraction of the coronary artery centerline from CCTA images is a challenging yet prerequisite task for many subsequent diagnoses, not only for FFR_{CT} estimation, but in daily clinical practice.

Typically, it is difficult to observe a tubular structure such as coronary arteries in a 3D CCTA volume directly. In standard clinical practice, the first step consists in extracting coronary artery centerlines manually from the volume, then doctors can examine a full coronary artery in a 2D curved planar reformation (CPR) image easily (see figure 2.2; as it will be mentioned in section 2.2.4, some automatic FFR_{CT} estimation methods leverage CPR as well).

Manual annotation of coronary artery centerlines and lumen radii is a time consuming

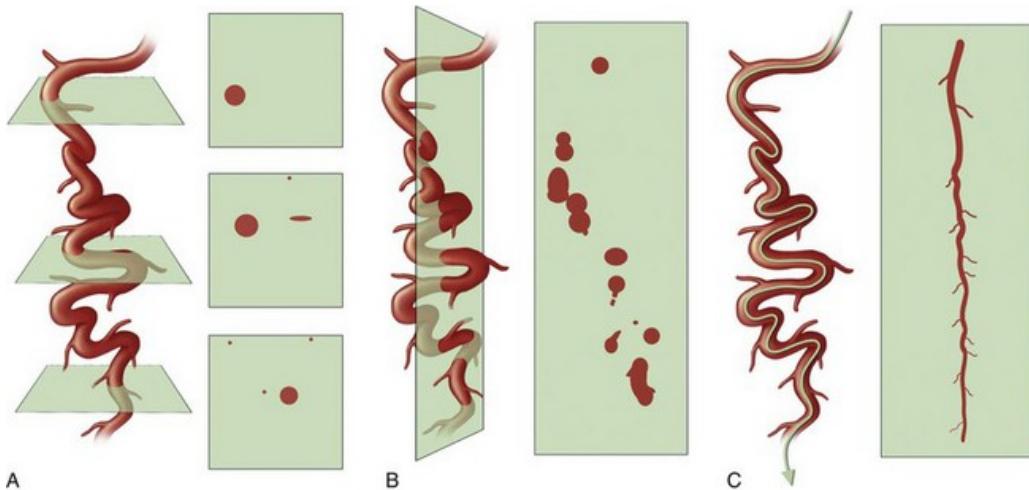


Figure 2.2: Schematic representation of the value of CPR in imaging complex vascular anatomy, both for humans and artificial intelligence. (A) Axial sections show only segments of the artery in cross section. (B) Multiplanar reconstruction in the orthogonal plane displays only various oblique cross sections of the artery. (C) CPR generated by centerline tracking captures and displays the entire artery. Source: radiologykey.com.

and burdensome task which has the potential to be automatized.⁶ The approaches for the extraction of coronary artery centerlines are categorized as: automatic requiring no user interaction at all, semi-automatic requiring one or more seed point(s) per artery, or interactive requiring multiple user interactions per artery.

Many (semi-) automatic methods that extract coronary artery centerlines have been proposed. Most of the methods can be divided into three categories:

- shortest path based [45, 97];
- segmentation based [85, 102, 108];
- tracking based [6, 12, 25, 99, 101, 106].

The first approach computes the shortest path in a vessel map between a start and an end point as the vessel centerline. Therefore, this method may need many interactions to extract the whole vessel tree.

The second approach, already introduced in Section 2.2.1, segments the vessels first and then extracts centerlines from the mask. Machine-learning approaches of this sort require large amount of accurately labelled training data. For instance, [85] used over 100 synthetic and 40 real data with mask annotation to train a convolutional neural network.

The third approach used a tracker based on hand-crafted features to iteratively search tubular structures such as retinal vessel [106], ridge-based vessel [101] and liver vessel [25].

⁶In [72] are reported approximately 30 pre-2009 papers that present and evaluate techniques for the centerline extraction of human coronary arteries in CCTA datasets.

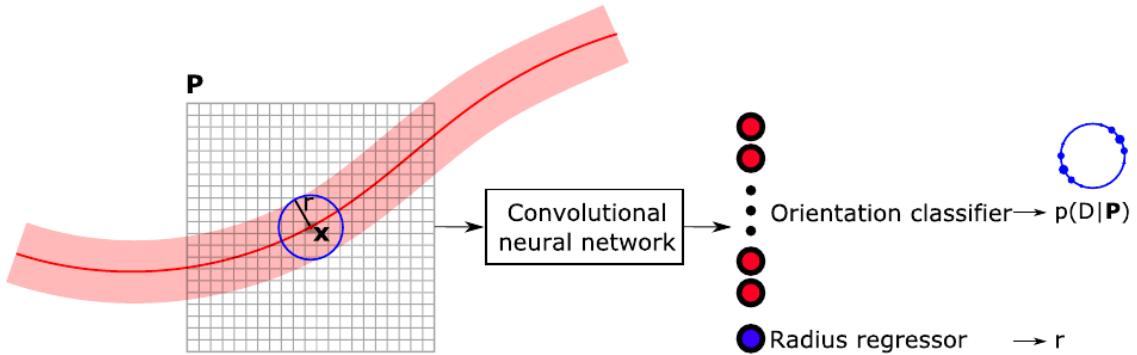


Figure 2.3: Overview of the method proposed by Wolternik et al. (3D-CNN tracker) [99]. At location x , an isotropic 3D patch P is extracted and used as input to a convolutional neural network (CNN). This CNN simultaneously determines a probability distribution over a discrete set of directions on a sphere (here shown as a blue circle), and an estimate r of the radius of the vessel (as defined in Appendix E and [72]). Source: Wolternik et al., 2018 [99].

The concept of a spherical tracker following the coronary arteries exist at least since 2012, when Yang et al. [103] proposed a 3D artery tracker method based on ray-casting inside a sphere, achieving, at the time, state-of-the-art performance.

More recently, in 2018 Wolternik et al. [99] proposed a convolutional neural network tracker (3D-CNN tracker) to interactively extract coronary artery centerlines and they obtained state-of-the-art performance. The tracker can simultaneously predict the proper direction and step-size using a single CNN. In the proposed method, a 3D dilated CNN is trained to predict the most likely direction and radius of an artery at any given point in a CCTA image based on a local image patch (figure 2.3). Starting from a single seed point placed anywhere in a coronary artery, a spherical tracker follows the vessel centerline in two directions using the predictions of the CNN. No image preprocessing is required, so that the process is guided solely by the local image values around the tracker's location. Termination of the tracker is guided by a stopping criterion based on the uncertainty of the direction classifier: the tracker terminates if the moving average over the last three steps of the entropy of the direction probability distribution crosses a threshold value of 0.9. Since automatic seeding was incorporated by means of another standalone neural network before the tree reconstruction step, the interactive procedure was rendered completely automatic (seeds placement could be also performed by considering [37]). The CNN was trained using manually annotated centerlines in training images (see Appendix E), and this method has been shown to work with high accuracy, with an average distance of down to 0.21 mm from reference centerline points during validation tests.

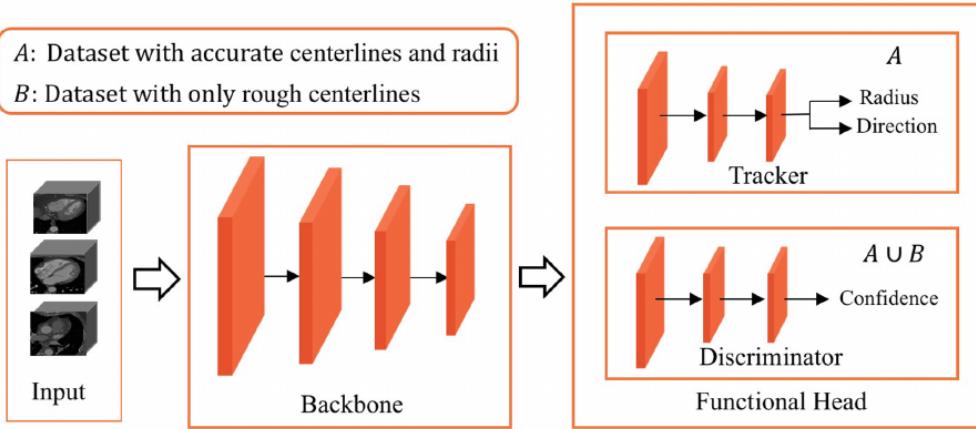


Figure 2.4: Detailed illustration of the DCAT method proposed by Yang et al. The backbone is a 3D CNN combined with first 4 convolutional layers used Wolternik’s work [99]. Dilation convolutional layers are used and dilation rate in layer 2 and 3 is 2 and 4, respectively. The tracker has three 3D convolutional layers and two outputs: the radius estimate r and direction probability distribution T . The discriminator has three 3D convolutional layers and outputs a confidence p . Source: Yang et al., 2019 [104].

However, the tracker may fall into other vessel-like objects during the automatic seeding process, because training samples are only extracted along the vessel centerlines and the stop criterion based on manual rules is inflexible. This approach also suffers from early-stopping issues, which means that many seeds need to be planted into a single artery to obtain the full tree with high enough reliability. In addition, training scans used in [99] require accurate centerline and radius annotations, which is a time-consuming work.⁷

One year later, Yang et al. [104] proposed an extension to the 3D-CNN tracker of Wolternik et al. [99] which solves, at least partially, the aforementioned problems. In this approach a discriminative coronary artery tracking method (denominated DCAT by the authors) is presented. The DCAT consists of a tracker and a discriminator. The 3D-CNN tracker identifies proper orientation and radius of the artery at each location, which is used to extract vessel-like objects. The discriminator is a binary classifier which provides a learning-based stop criterion during tracking (see figure 2.4). Training samples for the discriminator come from everywhere in the 3D CCTA volume, therefore it has ability to distinguish coronary arteries from other vessel-like objects. Training of the discriminator only needs coarsely labelled centerline annotations, that enables training

⁷It is fair to notice that open datasets already exist, such as the one from the MICCAI 2008 Coronary Artery Tracking Challenge (CAT08) [54, 55] or, better, the Rotterdam consensus-annotated dataset [72] described in Appendix 1. For industrial implementation purposes, it is true that requiring such precise annotations would be quite labour-intensive.

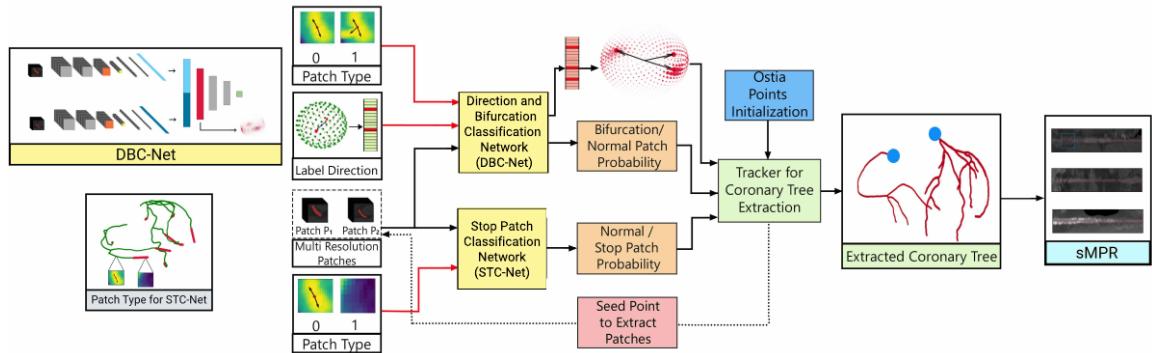


Figure 2.5: Overview of the AuCoTrack pipeline. Source: Salahuddin et al., 2021 [69].

the DCAT model with small amount of precisely labelled data (the same amount that would be needed for Wolternik’s 3D-CNN tracker approach [99]) and can instead leverage a large amount of coarsely labelled data, simultaneously. This can reduce annotation effort greatly considering the discriminator task. With this training schema, the model could be easily transferred to different domains with much less annotation workload. This approach obtained state-of-the-art performances both on the public Rotterdam dataset (Appendix E) and a privately collected dataset.

Another very recent approach extends the functionality of the work of Wolternik et al. [99] in a similar manner. Salahuddin et al. [69] proposed in 2021 a deep learning-based automatic tracker (denominated AuCoTrack by the authors) extending the 3D-CNN tracker. A multi-resolution 3D convolutional neural network is employed to simultaneously predict movement directions and detect bifurcations. Moreover, an automated artery endpoint detector is used to prevent premature termination of the tracking process.

This approach is interesting for many reasons. With respect to the 3D-CCN tracker of Wolternik et al. [99] they introduce a standalone CNN discriminator very similar to the one proposed by Yang et al.[104] to prevent premature termination. They introduced several novelties of their own as well. The involved neural networks operate locally on two isotropic 3D patches of size $19 \times 19 \times 19$ sampled at the two resolutions 0.5mm and 1.0mm whereas the previously discussed method used just one patch sampled at 0.5mm; this process allows to take full advantage of the resolution of the CCTA images which is nowadays usually smaller than 0.5mm per pixel. The approach also detects bifurcation points and properly integrates them into the tracking process. The algorithm does not rely on a segmentation network or any other automated method to generate multiple seed points per artery: a simple automated detection of two seeds (one for each coronary tree, see Appendix A) suffices and the proposed method is very robust towards variations of

those points. The ostia points⁸ obtained from a deformable shape model of the heart [21] are used to initialize the automatic coronary centerline extraction. This approach reportedly achieves better clinically relevant overlap than other automatic methods, including the ones previously considered in this document.

The work of Wolternick et al. [99] represent a milestone from which all of the most recent and best-performing state-of-the-art approaches take inspiration. Wolternik's results are still, at current date, used as a benchmark to compete against for other approaches and still remains one of the best approaches for coronary artery tree extraction.

The two most common subsequent steps to centerline extractions in the pipeline towards FFR estimation are (a) lumen segmentation and afterwards FFR estimation, or (b) direct FFR estimation with intrinsic lumen analysis. In the following section those methods relying on previous extraction of the centerline to perform lumen segmentation are considered.

2.2.3. Lumen segmentation subsequent to centerlines extraction

Many state-of-the-art FFR estimation approaches considered in this document do not require performing a lumen segmentation step and just require centerline extraction; some FFR estimation approaches actually embed lumen segmentation into their workflow not as an explicit extra step, but as a "hidden" step inside the procedure (oftentimes this step is really hidden inside the layers of the neural network used to analyse the lumen curved planar reformation volume).

Many older algorithms perform lumen segmentation directly without attempting a centerline extraction step beforehand, and they were usually based on standard image analysis techniques (3D edge detection, thresholding), some examples including region-based methods [39], morphology-based methods [11] and surface-based methods [105]. However, these methods pose many problems [90] and are not well suited for automatic operation with no supervision.

A more advanced approach includes graph-cuts methods for the segmentation of the arteries after centerline extraction. It works by defining a lumen edge energy function, and the graph-cuts method is used to minimise that energy, allowing for better lumen segmentation with respect to the previous approach. Some of these methods are summarized in [75].

⁸An *ostium* (from latin, plural *ostia*) is a mouth-like opening in a bodily part, such as in a blood vessel. The "entrance" and "exit" openings of a coronary artery in a CCTA images are referred to as *ostia points*; more precisely, the coronary ostium is the point where the coronary artery originates from the aorta. Refer to Appendix A for more details.

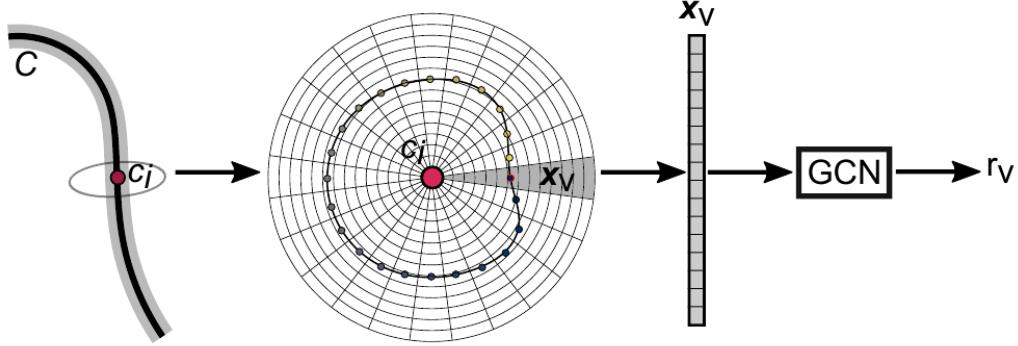


Figure 2.6: Overview of the 3D-CNN tracker + GCN method proposed by Wolternik et al. [98]. Given a coronary centerline C , rays are cast equiangularly and orthogonally to the centerline at each point c_i . Each ray angle corresponds to a vertex v in the luminal surface mesh of which the exact spatial location is determined by c_i , an angle ϕ_v , and the distance r_v . Image information along x_v is used by a GCN that combines this information with that of neighboring vertices to predict r_v . Source: Wolternik et al., 2019 [98].

Wan et al in 2019 [95] aimed to develop a method to automatically segment the coronary artery lumen based on anisotropic graph-cuts. This method follows the same basics of the previously mentioned approaches, however after image processing and rough centerline extraction, they segment the lumen by using anisotropic graph cuts. Although this method was proven to yield better results than the classic image analysis techniques, it still required a minor human intervention in the process, as some seed points have to be placed in the image for the algorithm to work. Another subtle but important drawback of methods based of graph cuts is that, if the algorithm is working on standard desktop computer with no particularly powerful hardware, it is very likely that the CCTA volume has to be reduced in size due to the massive amount of memory occupied by graph objects.

Deep learning based segmentation methods like U-Net and its variants have achieved promising results in medical image segmentation tasks [32, 34, 98], however these methods are not optimal for coronary artery segmentation due to the small size of the vessel and can cause poor segmentation performance [15] and often lead to reconstructed vascular rupture.

Novel lumen segmentation techniques focus on obtaining a surface mesh of the coronary arteries inner walls by leveraging deep learning techniques, starting from the extracted centerlines points. Two remarkable and recent examples are reported.

In 2019, Wolterink et al. [98] investigated graph convolutional networks for coronary artery segmentation in computed tomography angiographies, where vertices on the coro-

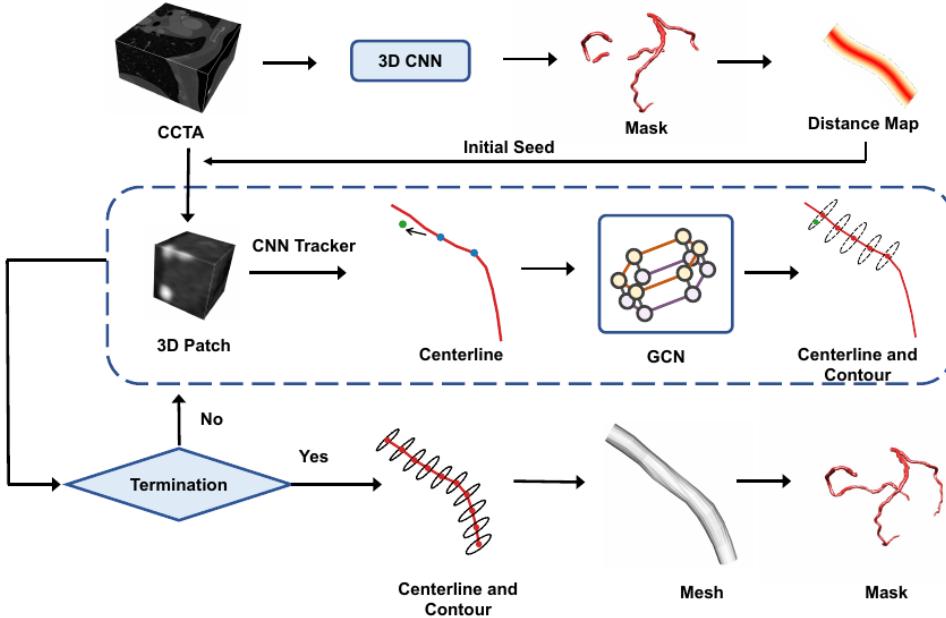


Figure 2.7: Pipeline of joint coronary centerline extraction and lumen segmentation method (3D-CNN tracker + GCN) proposed by Gao et al. Source: Gao et al., 2021 [26].

nary lumen surface mesh were considered as graph nodes and the locations of the tubular surface mesh vertices were directly optimized (see figure 2.6). They showed that such method significantly outperformed a baseline network that used only fully-connected layers on healthy subjects. Besides, the graph convolutional network used in their work is able to directly generate smooth-surface meshes without any post-processing step. This method is well suited for applications in which the texture of the luminar surface is of key importance. The mesh surface can be used to model the arteries topology for CFD methods, but could also be used for machine learning techniques exploiting topological analysis of the mesh surface.

In 2021, Gao et al. [26] proposed an automatic approach for joint centerline extraction and lumen segmentation from CCTA images using a hybrid of deep learning models. They introduced a hybrid model that tracks the centerline via the same 3D-CNN tracker used by Wolternik et al. [99] and captures the geometric shape prior that characterizes vascular relationship via graph convolutional network to obtain the complete coronary artery centerlines as well as the lumen contours, which can be converted into a lumen segmentation mask (see figure 2.7). Their approach has at least two distinctive strengths. Unlike U-Net, they model the geometry structure of vessels in lumen contour prediction via a graph convolutional network, rather than pixel-wise relationship. In contrast to

the previous segmentation method, which seriously depend on the accuracy of coronary artery centerline extraction, they segment artery and extract centerline jointly through a series of iterations. This approach has the effect of reconstructing longer artery segments with respect to the 3D-CNN tracker only, which is prone to drift and early stopping, and also increases the general performance.

It is interesting to notice that this last lumen segmentation method not only performs accurate lumen contour meshing, but also helps the tracker in the process.

Machine learning FFR estimation techniques solely or partially based on previously extracted 3D surface meshes represent an interesting ground to explore, as working with mesh surfaces should be much quicker than working with voxel-based volumes and shapes. In the next section, FFR_{CT} estimation techniques are discussed.

2.2.4. FFR estimation

Once the coronary artery centerlines have been correctly extracted (and/or lumen segmentation performed) the foundations are laid in order to estimate the FFR. FFR estimation from cardiac CTA is the final step of the considered context (CCTA - automatic centerline extraction - optional lumen segmentation and meshing - FFR_{CT} estimation), however it is not the primary concern of this document, which instead focuses on coronary artery centerlines extraction.

While it is possible to categorise segmentation and centerline extraction algorithms (see [72]), categorising machine learning-based FFR estimation methods is not trivial as it is still a novel challenge. The most interesting articles in the most recent literature are presented in chronological order. A study from May 2020 from Tesche and Gray [83] gives an overview of the technical background, clinical validation, and implementation of ML applications in FFR_{CT} .

In 2016, Itu et al. [36] presented the first remarkable machine-learning-based model for predicting FFR as an alternative to physics-based approaches. The model was trained on a large database of synthetically generated coronary anatomies, where the target values (the labels) were generated using the physics-based (CFD) model. The trained model predicts FFR at each point along the centerline of the coronary tree, and its performance was assessed by comparing the predictions against both physics-based computations and invasively measured FFR for 87 patients and 125 lesions in total. Correlation between machine-learning and physics-based predictions was excellent (0.9994 , $P < 0.001$), and no systematic bias was found. An invasive $\text{FFR} \leq 0.80$ was found in 38 lesions out of 125 and was predicted by the machine-learning algorithm with a sensitivity of 81.6%,

a specificity of 83.9%, and an accuracy of 83.2%⁹, while the correlation was 0.729 ($P < 0.001$). Compared with the physics-based computation, average execution time was reduced by more than 80 times, leading to near real-time assessment of FFR. This first attempt from Itu et al. showed very promising results and marked the beginning of a more extensive research on the matter.

In a study evaluated in 2019, Wang et al. [96] developed the DEEPVESSEL-FFR platform using the emerging deep learning techniques to compute the FFR_{CT} in five minutes. Predictions by means of the DEEPVESSEL-FFR along the coronary artery tree is performed by a deep learning algorithm architecture constructed with multilevel neural network (MLNN) and bi-directional recursive neural network (BRNN). The network is arranged in two phases. The first phase consists in the interpretation of features from the input volume by using a MLNN. The second phase is comprised of a BRNN that receives the features sequence to the next layer. The FFR_{CT} distribution in the coronary artery tree is found. During the training, ground truths labels were generated by solving the Navier-Stokes equations with invasive FFR as a reference. This CFD step was needed to synthetically create the ground truth data (synthetic labels) along the entirety of the centerline of the vessels, starting from a single or a few actual invasive FFR measurements. This method does not guarantee that the network is being trained on 100% correct data, however their test results performed on 68 patients showed good statistics with regards to the proposed approach. By using the CFD for training, they "incapsulated" it into the neural network, allowing a merged approach with the benefits of the superior computational speed and generalisation capabilities of a neural network.

Meanwhile in 2019, Zreik et al. [111] presented a method for automatic and non-invasive detection of patients requiring invasive coronary angiography, employing deep unsupervised analysis of complete coronary arteries in CCTA images. The centerlines of the coronary arteries were automatically extracted and used to reconstruct straightened multi-planar reformatted (MPR, similar to previously discussed CPR) images of the coronary arteries (see figures 2.8,2.9). Then, the MPR volume of a complete coronary artery is encoded into a fixed number of features using two disjoint convolutional autoencoders, applied sequentially: a 3D variational convolutional autoencoder (3D-VCAE), that spatially encodes local sub-volumes of the coronary artery, and a 1D convolutional autoencoder (1D-CAE), that sequentially encodes the features of the complete artery. Then,

⁹Sensitivity, specificity and accuracy are widely used statistics to describe a diagnostic test. In particular, they are used to quantify how good and reliable a test is. *Sensitivity* evaluates how good the test is at detecting a positive disease. *Specificity* estimates how likely patients without disease can be correctly ruled out. *Accuracy* measures how correct a diagnostic test identifies and excludes a given condition. [109].

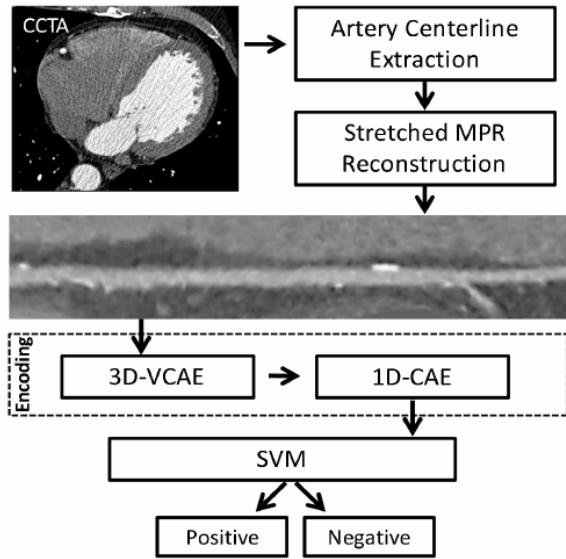


Figure 2.8: Pipeline of the method proposed by Zreik et al. In a CCTA scan, the centerlines of the coronary arteries are extracted and used to reconstruct straightened multi-planar reformatted (MPR) images of the coronary arteries. Then, an unsupervised analysis is performed, where the MPR volume of a complete artery is encoded into a fixed number of features (see adjacent image). Then, the final extracted encodings are employed in a supervised fashion to classify arteries according to the need of further invasive evaluation to establish the need of intervention, using a support vector machine (SVM) classifier.

Source: Zreik et al., 2018 [111].

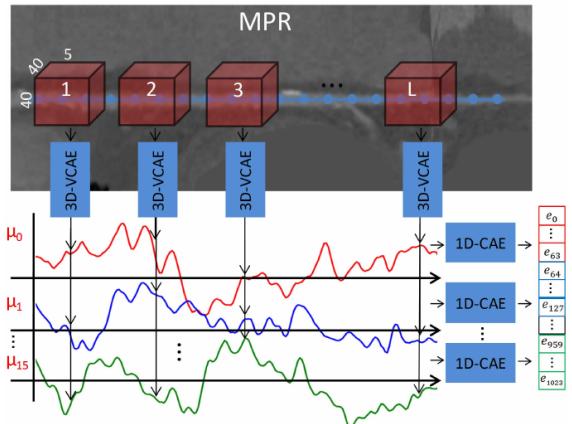


Figure 2.9: The MPR volume of a complete artery is encoded into a fixed number of features (or encodings) using two disjoint convolutional autoencoders, applied sequentially: a 3D variational convolutional autoencoder (3D-VCAE) that spatially encodes local sub-volumes of the coronary artery, and a 1D convolutional autoencoder (1D-CAE) that sequentially encodes the encodings of the complete artery. Source: Zreik et al., 2018 [111].

the final extracted features are used to classify arteries according to the need of further invasive evaluation, using a support vector machine (SVM) classifier.

Zreik et al. in early 2021 presented a method for non-invasive detection of patients with functionally significant coronary artery stenosis, combining analysis of the coronary artery tree and the left ventricular myocardium in CCTA images [112]. Coronary arteries are encoded by two disjoint convolutional autoencoders and the left ventricular myocardium is characterized by a CNN and a convolutional autoencoder. Thereafter, using the ex-

tracted encodings of all coronary arteries and the left ventricular myocardium, patients are classified according to the presence of functionally significant stenosis, as defined by the invasively measured FFR. To handle the varying number of coronary arteries in a patient, the classification is formulated as a multiple instance learning problem and is performed using an attention-based neural network. Formulating the analysis of all coronary arteries in the coronary tree as a multiple instance learning problem was never presented before. For training, they retrospectively collected CCTA scans of 126 patients who underwent invasive FFR measurements, to determine the functional significance of coronary stenoses. Cross-validation experiments showed that the combined analysis outperformed the analysis of the coronary arteries [111] or the left ventricular myocardium [110] alone.

The last two works of Zreik et al. (of which Wolterik is reported as a co-author) represent a very interesting workflow which should be further investigated, especially the combined analysis approach. For the combined workflow to work, accurate image segmentation and registration is needed. While Zreik and Wolternik proposed their own method in the reported approaches, Sinclair et al. (with Michiel Schaap¹⁰ as a co-author) [78] recently proposed a very versatile approach for joint segmentation, registration and atlas construction with image-and-spatial transformer networks. This approach is very promising to segment and register both tubular structures and the myocardium.

In contrast to purely machine-learning based methods, in 2020 Papamanolis, Schaap et al. [64] proposed a hybrid machine-learning/simulation based approach involving a multiscale patient-specific model enabling blood flow simulation from large coronary arteries to myocardial tissue (see figure 2.10).

Patient vasculatures are segmented from coronary computed tomography angiography data and extended from the image-based model down to the arteriole level using a space-filling forest of synthetic trees. Blood flow is modeled by coupling a 1D model of the coronary arteries to a single-compartment Darcy myocardium model. Simulated results on five patients with non-obstructive coronary artery disease compare overall well PET exam data for both resting and hyperaemic conditions. Results on a patient with severe obstructive disease show that the approach can link coronary artery narrowing with impaired myocardial blood flow, demonstrating the model's ability to predict myocardial regions with perfusion deficit.

Since the state-of-the-art analysis in this document is quite dense with information, the next section offers a summary of the analysed pipeline to estimate the FFR_{CT} in the

¹⁰Michiel Schaap is a very active medical image researcher, alumnus of the Biomedical Imaging Group Rotterdam and currently working at the Biomedical Image Analysis Group, Imperial College London and for HeartFlow®, USA. Author of [72] and co-author of other cited works.

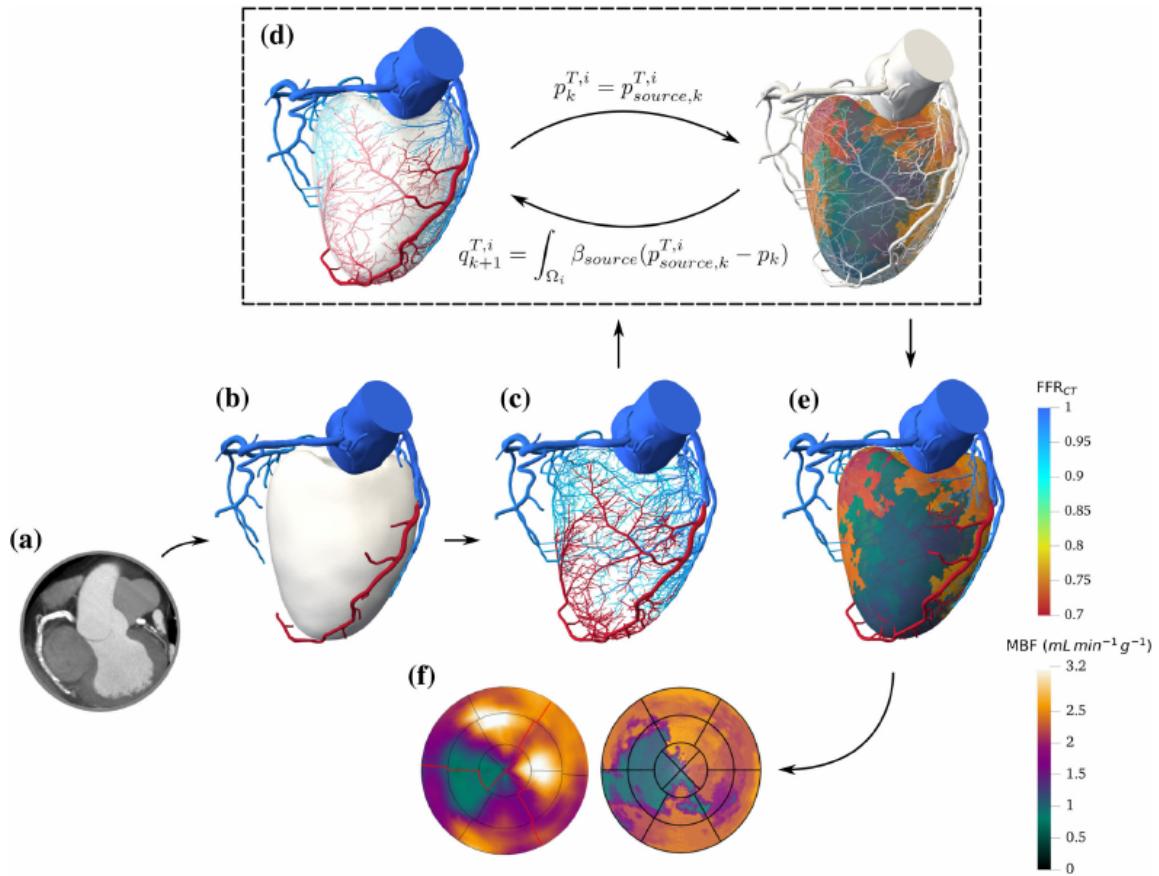


Figure 2.10: Illustration of the pipeline for a patient with disease in the left anterior descending coronary artery. (a) CT imaging data. (b) Segmented geometry and FFRCT analysis. (c) FFRCT results in segmented and synthetic vasculature. (d) Illustration of the coupling loop, demonstrating quantities exchanged between the coronary model (left) and the myocardium model (right) at coupling iteration k . (e) Hyperemic MBF for coupled model. (f) Comparison of simulated (right) and PET (left) hyperaemic perfusion maps. Source: Papamanolis et al., 2020 [64].

coronary artery tree. Thereafter, a subsection of the general pipeline is analysed and discussed as the main matter of this thesis.

2.3. Main matter declaration and context

This chapter presented a comprehensive analysis of the state-of-the-art regarding all the main steps involved in the estimation of the FFR index from CCTA images, summarised in figure 2.11.

The first step in the pipeline is the coronary arteries centerlines extraction step which, as discussed, can be codependent on lumen segmentation.

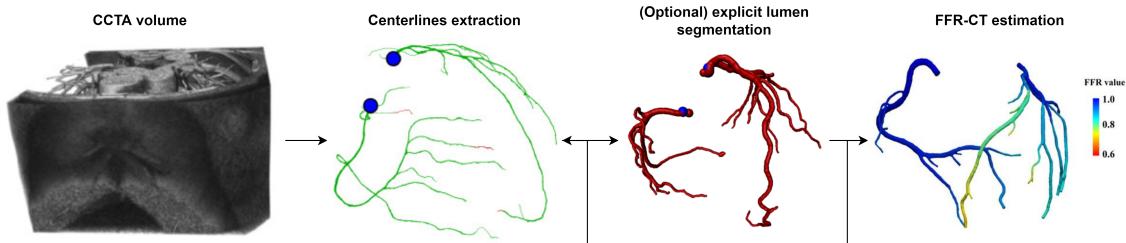


Figure 2.11: Overview of the end-to-end pipeline to estimate the FFR_{CT} index.

The lumen segmentation step is somehow always performed, but many FFR_{CT} estimation methods do not rely on an explicitly segmented lumen; rather, they incorporate this step into the machine-learning stack. On the other hand, lumen segmentation is a mandatory step for FFR_{CT} estimation methods based on computational fluid-dynamic simulations. Lastly, the latest FFR_{CT} estimation methods were presented. The presented methods can take many features as input: many require just the extracted coronary arteries centerlines to perform FFR_{CT} estimation along the artery tree, while some others rely also on other features, such as the entire myocardium.

The main matter of this thesis is concerned with the geometric reconstruction of the arteries of the coronary artery tree which are considered "clinically relevant" by the medical community [49, 68, 72]: while the LCA and RCA (left main and right coronary arteries) and their branches can be considered clinically relevant almost in their entirety, their sub-branches can have a lumen equivalent diameter of far less than 1.5mm, thus making tracking sub-branches a hard task, which is not required in the majority of situations. The centerline extraction task will be considered in the context of the FFR_{CT} estimation pipeline, however fully-automatic centerlines extraction has many other uses in clinical practice (section 2.2.2).

The main objectives of this document are the following.

1. To provide an extensive analysis of the current state-of-the-art approaches regarding the whole FFR_{CT} estimation pipeline.
2. To re-implement the 3D-CNN tracker proposed by Wolternik et al. [99], to show that it is a valid starting point for the creation of new trackers, and explore new ways to improve its tracking capabilities and extension, such as new neural network architectures, new training approaches, and new trackers and functionalities.
3. Moreover, the final coronary artery tree can be obtained in the form of a connected graph starting from the cloud of points generated by the tracker. This procedure is

performed in a fully automatic and unsupervised way, which is something that no paper reported in this document proposes.

4. To create a test suite for the explored algorithms. The different neural networks at the core of the tracker will be tested locally on single patch predictions, while the tracker itself and the graph reconstruction algorithm will be tested globally with respect to the reference centerlines, similarly to how it is described in [72].

In the next chapter the second and third main objective of this document will be discussed, starting by the structure of Wolternik's CNN [99]. Some new neural network structures will be discussed as well. Following, the dataset used to train the networks, the training conditions, and the structure of the tracker will be discussed.

3 | Methods: the 3D-CNN Tracker and its descendants

In this chapter, two modifications are proposed on both the CNN architecture and the tracking algorithm of Wolternik et al. [99]. Their 3D-CNN tracker is analysed, reimplemented, and it is shown that the proposed solutions both surpass the performances of their 3D-CNN tracker tested on the same evaluation framework.

Some post-processing steps are presented to automatically classify the obtained cloud of centerline points into the two main artery branches (left and right), detect all the endpoints and, through an oriented graph, connect all the points to form a unique, oriented and topologically consistent vessel tree which also allows to segment all the different centerline branches, completely automatically.

In daily clinical practice, it is much preferred to perform non-invasive assessment of coronary lesions rather than risky invasive procedures, both for visual assessment through multi-planar reconstructions and, lately, for automatic FFR index estimation. Cardiac computed tomography imagery is preferred due to its ability to precisely capture many features and details useful for a complete assessment of the severity of a lesion, which facilitate stenosis detection and plaque identification. A prerequisite step for both humans and machines to perform such assessments is the extraction of the luminal geometric centerline of the arteries under scrutiny, task that is tedious and time-consuming for human operators.

Automatic coronary arteries centerline extraction has been a hot topic of research in CCTA showing vascular structures. Minimal cost paths methods may suffer from shortcuts between different points on the centerline, centerline extraction after lumen segmentation (discussed in section 2.2.3) require the analysis of the whole volume thus being relatively time consuming, and methods based on idealised tubular models, hand-crafted vesselness filters and shape priors suffer greatly when exceptions such as bifurcations and stenotic areas are presented. Prior machine-learning methods ([29, 73, 80]) do not relied on hand-crafted vessel filters or models, however they required an intermediate feature

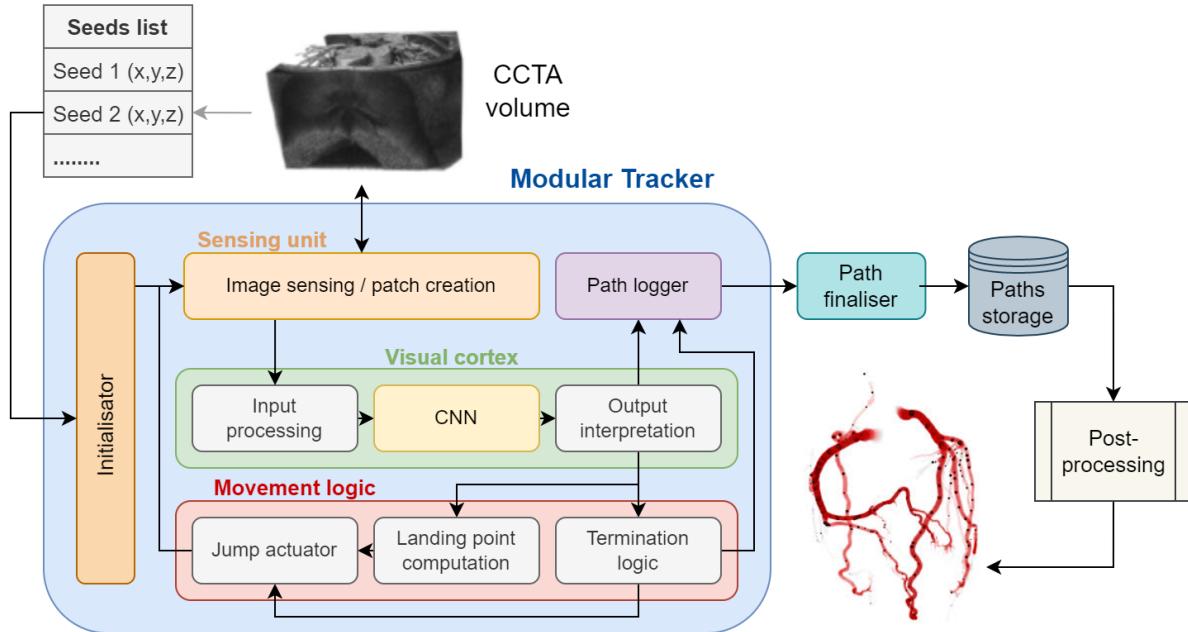


Figure 3.1: Representation of the tracker and its modules (boxes) and interfaces (arrows). Any number of additional modules could be added to the tracker and connected to any other component through interfaces. Moreover, any network can be plugged in in the middle of the "visual cortex", provided that "input processing" and "output interpretation" are extended accordingly. For what concerns the proposed networks, any network can be plugged in and out at will without needing any other intervention on the tracker.

representation of the data and evaluation at multiple voxel locations to obtain the vessel orientation at a single point, which is time consuming. On the other hand convolutional neural networks have demonstrated the ability to derive useful features directly from image data in a wide range of applications including medical image analysis tasks, which suggests that CNNs could also be used to determine vessel centerlines.

With CNNs all information is extracted directly from the image, and no intermediate hand-crafted vesselness representation is required. The method can be trained with manually annotated reference centerlines in a limited set of training images since each image contains plenty of training samples.

A tracker approach also has another positive feature. Differently from many other approaches which analyse the image on the whole in a strictly procedural way, a tracker can be implemented as a very modular piece of software through object-oriented programming: once the basic functionality of a tracker are implemented, such as moving from one point to the other in a consistent way, its functionalities can be easily extended through inheritance simply by "plugging in" new methods or overwriting existing ones. A modular

tracker can be quickly set up to support any kind of artificial neural networks (or even other models) and can be quickly adapted to any changes in its inputs and outputs while prototyping and testing new solutions, without disrupting the whole ecosystem. In figure 3.1 is represented a scheme of the modular architecture of the tracker and its components.

In the next section the neural networks are presented and the underlying principles explained. In the following one, the tracking algorithm is explained in details and different trackers are presented.

3.1. Neural networks, the core of the tracker

Prior continuing with the discussion regarding the specific network architectures in use, the main characteristics of convolutional neural networks are explored and the main features inspected. An introduction to the general theory of machine learning is presented in Appendix C.

As it was stated in the state of the art analysis, prior to CNNs manual, time-consuming feature extraction methods were used to identify objects and features in images. Semi-automatic procedures were time consuming too, because based on the same principles of manual feature extraction. However, convolutional neural networks now provide a more scalable approach to image classification and object recognition tasks by leveraging two principles, matrix multiplication and convolution operator, to learn to identify patterns within an image.

The architecture of a CNN is similar to that of the connectivity pattern of neurons in the human brain and was inspired by the organization of the visual cortex¹. Individual neurons respond to stimuli only in a restricted region of the visual field known as the "receptive field" of the neuron. A collection of such fields overlap to cover the entire visual area. Similarly, a feature detector (also known as a kernel or a filter) moves across the receptive fields of the image, "checking" if the feature is present. This process is known as a convolution.

In the case of patches (sub-volumes, figure 3.2) of CCTA volumes, the feature detector is a three-dimensional array of trainable weights F , which through training learn to represent important parts and features of the image. While they can vary in size, a typical filter size is $3 \times 3 \times 3$ (figure 3.2). The size of the receptive field is determined by the size of the kernel and by the dilation, which is the amount of pixels to go from one "sensor" pixels of the kernel to the next one (figure 3.3). The filter is then applied to a sub-volume of the

¹More resources by Puttatida Mahapattanakul at:
becominghuman.ai/from-human-vision-to-computer-vision-how-far-off-are-we-part1-3-b35d37a196a4.

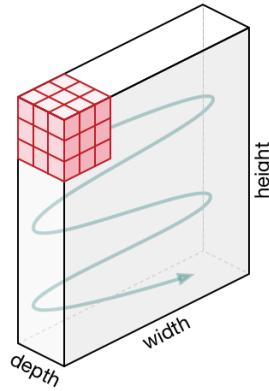


Figure 3.2: An example of 3D kernel and its movement across a portion ov volume. Source: towardsdatascience.com.

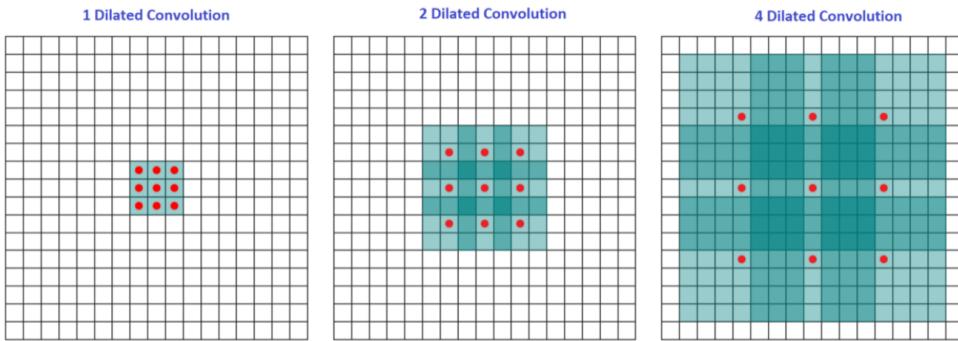


Figure 3.3: An example of 2D kernel dilation, which increases the receptive field. Source: Yu, Koltun, 2016 [107]. Another source: github.com/vdumoulin/conv_arithmetic/blob/master/gif/dilation.gif.

image V_{input} of the same dimension of the filter, and a dot product is calculated between the input pixels and the filter:

$$V_{input} \cdot F = \sum_{i,j,k} v_{i,j,k} f_{i,j,k}$$

The result is then fed into an output array. Afterwards, the filter shifts by a stride (typically 1, 2, or 3), repeating the process until the kernel has swept across the entire image (figure 3.4)². The final output from the series of dot products from the input and the filter is known as a feature map. Many different filters can be swept across the image, creating many different feature maps (in figure 3.4 these are the b_i^l collection of feature maps). The number of applied filters (which is also the number of obtained feature maps)

²Other image sources concerning CNN basic convolution operation: https://github.com/vdumoulin/conv_arithmetic/tree/master/gif.

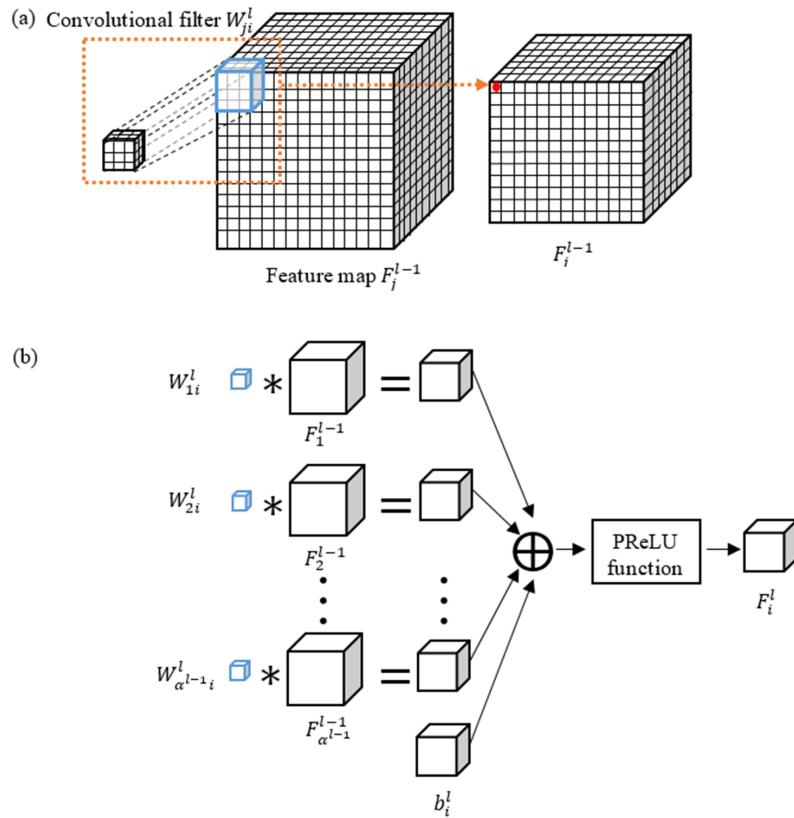


Figure 3.4: An example of a 3D convolution operation. (a) 3D convolution of a input volume/feature map with a filter. (b) Generation of the i_{th} feature map in the next layer. Source: Liu et al, 2017, DOI: <https://doi.org/10.1371/journal.pone.0185844>.

is the number of output channels.

The objective of the convolution operation is to extract higher-level features from the input, such as edges or complex shapes, from the input image. Another convolution layer can follow the initial convolution layer. When this happens, the structure of the CNN can become hierarchical as the later layers can see the pixels within the receptive fields of prior feature map. Conventionally, the first layer is responsible for capturing the low-level features such as edges, gradients and orientation. With added layers, the architecture adapts to the high-level features as well, such as larger elements or shapes of the object, creating a network capable of having a more comprehensive understanding of images in the dataset.

There are a number of tuneable hyperparameters which affect how each layer, and thus the network, is built and behaves. These include:

- Number of input channels: it is the number of same-dimension images passed to the layer. For a 2D RGB image, there can be 3 input channels to the 2D convulsive

filter; for a 3D grayscale volume, there will be one channel to the 3D filter. If from the previous layer are obtained 10 3D feature maps from 10 different filters, then the number of input channels will be 10.

- Number of filters/output channels: it affects the depth of the output. For example, three distinct filters would yield three different feature maps per each input channel, creating a depth of three.
- Kernel size: it is the size of the convolving kernel (the 3D array of weights). In figure 3.2, the red kernel is 3x3x3.
- Stride: it is the distance, or number of pixels, that the kernel moves over the input matrix. A higher stride (2 or more) yields a smaller output feature map, however such values are rare.
- Dilation: spacing between the kernel elements (figure 3.3). It increases the receptive field exponentially by increasing the number of parameters linearly [107].
- Padding: it is the added volume to the 3D input patch so that the center of the kernel can sweep all the pixels of the original image/volume.

The architecture of convolutional neural networks, by mimicking the visual cortex of animals, allow the network to autonomously learn the most significant features that are best suited for the intended task. The concept behind the whole procedure is similar to using standard image analysis filters, shape priors and models, however it cannot be shown that a certain set of standard filters (i.e. Sobel, etc) or shape priors (tubular structures) is "optimal" to accomplish a certain task. In the case of CNNs the model has total freedom over the definition of such filters and features, and can learn the best ones task-wise by performing optimisation on the whole feature complexity stack, from low level features to the highest level ones.

In the case of coronary artery tracking, the CNN learns to recognise the best set of features at different complexity levels to perform predictions about the orientation of the coronary artery with respect to the center of the image patch in two possible directions, forward and backwards.

The structure of the CNN proposed by Wolternik et al. [99] is discussed next, as it represent the most solid starting point for possible expansions yielding state-of-the-art performances.

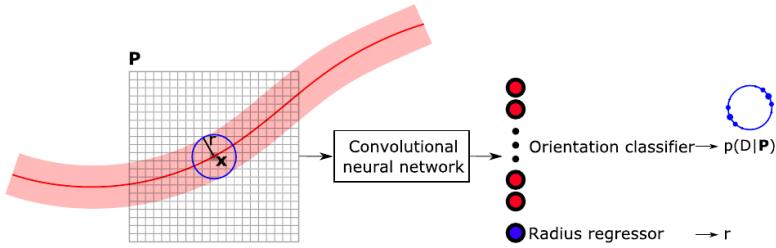


Figure 3.5: I/O scheme of Wolternik's CNN [99]. Input represented in 2D for clarity, the patch P is 3D, and a square pixel in the image represents a cubic voxel in the CCTA. Source: Wolternik et al., 2018 [99].

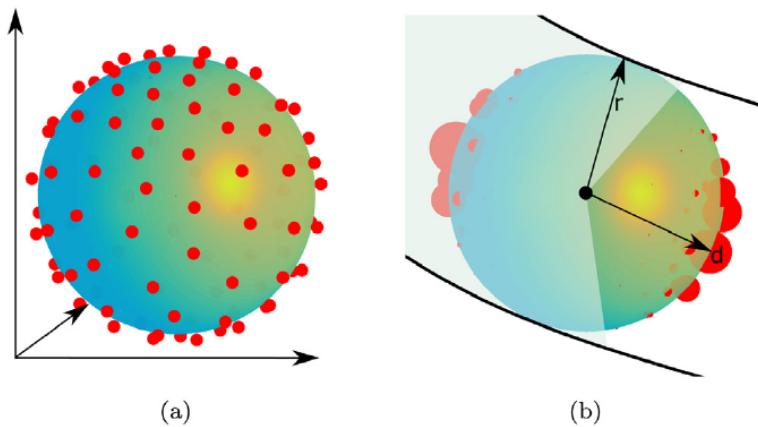


Figure 3.6: The set of possible directions distributed on a sphere (also called "shell") and the probability distribution on the shell. Source: Wolternik et al., 2018 [99].

3.1.1. Wolternik's CNN

Wolternik et al. [99] proposed in 2018 a CNN based method to determine the orientation and radius of a coronary artery at a location x in an image I , based on a 3D isotropic image patch P and using a single CNN. The output layer of the CNN consists of classification nodes that determine a probability distribution over some fixed tracking directions D , and a regression node that determines the radius R (r in figure 3.5) of the vessel at the point x .

The possible directions D are distributed on a sphere, where each point corresponds to an output class of the network (figure 3.6, a). Orientation determination is posed as a classification problem rather than a regression problem, so that the CNN may return a posterior probability distribution with multiple local maxima during tracking (figure 3.6, b). In contrast to the potential directions, the radius can be estimated by a single scalar value.

The 3D image patch P is a portion of I , which is the re-sampled original CCTA image I_{orig} to have an isotropic voxel spacing of v mm/voxel. P is of $w \times w \times w$ voxels, centered at x . The values for w and v together determine the input resolution and physical receptive field of the CNN in real world dimensions, and may be chosen before training depending on vessel caliber. The reference implementation uses $w = 19$ voxels and $v = 0.5$ mm, corresponding to a physical patch side of 9.5mm, sufficient to cover even the widest portions of the coronary arteries.

The structure of the convolution layer and their hyperparameters are shown in table 3.1.

Layer	1	2	3	4	5	6	7
Kernel width	3	3	3	3	3	1	1
Dilation	1	1	2	4	1	1	1
Channels	32	32	32	32	64	64	$ D +1$
Field width	3	5	9	17	19	19	19

Table 3.1: Structure of Wolternik’s CNN [99]. For each layer (Layer), the convolution kernel width (Kernel width), the dilation level (Dilation), the number of output channels (Channels), and the receptive field at that layer (Field width, voxels) are listed. The number of the final output channels is equal to the number of potential directions in D , plus one channel for radius estimation. Total number of trainable parameters: $\sim 176'500$ with $|D|=500$.

The CNN architecture contains a stack of convolution layers that aggregate features over multiple scales with convolution kernels that have an increasing level of dilation. As the level of dilation increases (light blue columns in table 3.1), the receptive field at each layer increases as well, but the number of trainable parameters per kernel stays the same at $3 \times 3 \times 3 = 27$. Hence, the receptive field grows exponentially from 3 to 7 to 15, but the number of trainable parameters increases linearly ([107]). Reducing the number of parameters generally helps preventing overfitting of 3D CNNs and allows fast processing. In addition, the proposed network does not perform down-sampling from one layer to the next, as it is common in CNN applications, in order to keep full resolution feature maps. To convey more information about an increasing receptive field, the network gets wider towards the output layers (the number of output channels increases).

The classification nodes are combined through a softmax activation layer. In contrast to the potential directions, the radius can be estimated by a single scalar value. Hence, this value is estimated using regression with a linear activation function. The $|D|$ classification nodes and the single regression node together form the output layer of the CNN.

Aside from the softmax classification and linear regression nodes in the output layer, all nodes in the network use rectified linear units (ReLUs) as activation function. Batch normalization is applied in each layer of the network. The last two fully connected layers are implemented as $1 \times 1 \times 1$ convolutions (light yellow in table 3.1), which have exactly the same effect.

The different direction classes D are created by creating a spherical shell using the Fibonacci lattice mapped onto a sphere:

$$(x_i, y_i) = \left(\frac{i}{\alpha} \% 1, \frac{i}{n} \right) \text{ for } i = 0, \dots, n - 1, \alpha = \frac{1 + \sqrt{5}}{2}$$

$$(x, y) \longrightarrow (\theta, \phi) : (2\pi x, \arccos(1 - 2y))$$

$$(\theta, \phi) \longrightarrow (x, y, z) : (\cos(\theta)\sin(\phi), \sin(\theta)\sin(\phi), \cos(\phi))$$

where the $(\cdot) \% 1$ operator denotes the fractional part of the argument.

There exist some more efficient ways of defining evenly distributed points on a sphere³, however the Fibonacci lattice is more than sufficient for this application.

This model and the followings are implemented in Python using the PyTorch library (Paszke et al., 2017 [66]). PyTorch is a low-level library which has already implemented the basic modules to build the network exactly as described in the paper: the seven layers are stacked considering the hyperparameters, and each layer is made of a stack of `Conv3d()` followed by a `BatchNorm3d()` followed by a `ReLU()` activation.

3.1.2. First proposed network: input-layer-split

The neural network and tracker proposed by Wolternik [99] shows many wise considerations, for instance the use of directional classes to transform a regression problem into a classification problem, or also the dilation of two middle convolution layers to increase the receptive field and keeping a lower number of parameters which allows to not perform pooling between the layers and keeping the features at their maximum resolution. The same feature extraction filters are used to both classify direction and estimate the radius, which makes the network compact but requires that the shared weight be adjusted to accommodate both tasks. Sharing weights to this extent might lead to an undersized network which has to compromise too much to perform both tasks well enough.

This possibility led to the investigation of new network structures which leave more freedom to each task. Being direction and radius estimation quite different tasks, it might

³For further insights:
<http://extremelearning.com.au/how-to-evenly-distribute-points-on-a-sphere-more-effectively-than-the-canonical-fibonacci-lattice/>.

be beneficial to split the network, if not in its entirety, at some point in its layers. This should allow the network to learn some task-specific features representations which could be independent from each other, and thus have more focused learning capabilities at a cost of an increased weight of the network and computational effort.

It makes sense to keep the first few layers in common to jointly extract low-level features, and then splitting the network at some point. However, a first tentative is made by splitting the network at the first layer, right after the input, to feel the ground and understand if this is a viable direction. This is equivalent to apply two distinct networks to the same input, which doubles the size of the network (for a total of more than 340k trainable parameters) and presumably increases computation times, but allows the whole network to learn the best set of features to perform the direction estimation and radious estimation tasks, independently.

Results (chapter 4) show that this path is viable and should be further investigated. In fact, the network has further room of improvement; no parameters are shared between the two tasks, which means that the network is far heavier than the baseline and a little more prone to overfitting. It also means that, while training, each branch of the network can rely just on information either on the radious or the direction to learn.

The network discussed next proposes some possible improvements both in network's weight, speed, and parameters sharing.

3.1.3. Second proposed network: midsplit-4

The proposed network is built on the basis of Wolternik's [99] up until the fourth layer, after which the network is split in two independent branches, one concerning purely the direction classification task, the other the radious estimation.

The shared layers serves mainly three purposes. Most obviously, the network is lighter than the input-layer-split network, with a total of $\sim 236'300$ trainable parameters – just 33% more than Wolternik's network [99] and more than 33% lighter than the input-layer-split network.

The main purpose of splitting the network after the first 4 layers is to let the network learn the basic geometrical features useful for both tasks, and let the high-level features adjust to each task independently.

This approach is common in convolutional neural network for image processing, where the basic features such as edges and boundaries are expected to be the same in any context, while the high-level features extracted from the low level ones are expected to change with respect to the task. For example, every human eye can discern a flower from its

background and understand its basic features, however a few, specifically trained humans can classify a flower at first sight. In fact, the visual cortex of mammalian and birds is divided in "simple cells" processing the electric stimuli from the retina, and "complex cells" which, instead (and sometimes at the same time), process the stimuli from the simple cells [50]. Of course biology is far more complex than this simple subdivision of types and roles, which are still under investigation by neuro-scientists; however, this is of inspiration to try and create a subdivision higher up in the network, where the first four layers play the role of the "simple cells", the remaining split convolution layers the role of the "complex cells" and the fully connected layers serve as the final processing unit.

The structure of the midsplit-4 network is shown in table 3.2. The output from layer 4 is the input of both subsequent sections of the network.

Layer	1	2	3	4	5	6	7	
Kernel width (K)	3	3	3	3	3	1	1	(K)
Dilation (D)	1	1	2	4	1	1	1	(D) Direction
Channels (C)	32	32	32	32	64	64	D	(C) classification
Field width (F)	3	5	9	17	19	19	19	(F)
					3	1	1	(K)
					1	1	1	(D) Radious
					64	64	1	(C) regression
					19	19	19	(F)

Table 3.2: Structure of the midsplit-4 network. For each layer (Layer), the hyperparameters of the convolution filters are listed. After layer 4, the network splits into two independent networks. Total number of trainable parameters: $\sim 236'300$ with $|D|=500$.

The goal of the proposed networks is to estimate the best direction and radious to move forward into the vessel, which means that what is obtained from the network is a displacement vector $\vec{\Delta} = \vec{D}_{best} \cdot R_{est}$ that the tracker uses to jump from the current pixel position to the next position. In this context, and also in the context considered in this document (fully-automatic FFR_{CT} estimation), the importance of the radious estimation is shifted more towards the creation of the displacement vector rather than a faithful anatomic estimation of the vessel luminal dimension⁴.

Because of this fact, the possibility of adding knowledge regarding the direction predicted by one branch of the network into the other branch of the network is explored in the next

⁴This is particularly true because the approximation of a perfectly tubular structure of the vessel is a gross approximation which is more often wrong than not. In the reference Rotterdam dataset annotations, the radious is just a measure of the equivalent area of the luminal section of the artery [72], which means that inferring liminal geometry from such measure is bad practice and would result in mostly wrong estimations.

network.

3.1.4. Third proposed network: midsplit-4-FD

The network midsplit-4-FD (feedback on directions) tries to enhance the displacement vector's modulus R estimation by inserting a feedback loop into the network, possibly sacrificing a little in the direction estimation precision. The rationale behind this choice is that, once the most likely directions are known, the displacement vector's modulus R can be estimated more precisely by using the knowledge about the direction distribution.

The new network structure is represented in figure 3.7.

Another fully connected layer is introduced in the radius estimation branch, to let the regression task better handle eventual non-linearities between the direction classes and the extracted features from previous convolutions.

In the next section, the dataset is presented as well as the original training strategy by Wolternik et al. [99] Some attempts at modifying the original method are discussed.

3.2. Dataset and training strategies

The neural networks were trained and tested on 8 of the 32 CCTA images from a publicly available evaluation framework for coronary centerline extraction, the MICCAI 2008 Coronary Artery Tracking Challenge (CAT08) which is part of the Rotterdam Coronary Artery Evaluation Framework ([72]). Images in CAT08 were acquired on two different 64-slice CT scanners. Images were reconstructed to a mean voxel size of $0.32 \times 0.32 \times 0.4 \text{ mm}^3$.

In each scan, the centerline and radius of four major coronary arteries were manually annotated in a consensus reading by three experts. The CAT08 organizers separated the 32 CCTA images into a training set and a test set based on subjective image quality and coronary calcium burden. The training set consists of 8 CCTA images with reference annotations for the centerline location and radius, and the test set consists of 24 CCTA images for which no reference annotations were released to the public.

The evaluation framework and the composition of the dataset is further discussed in Appendix E and the detailed description of scan acquisition, reconstruction, and the centerline annotation protocol is provided in the work of Schaap et al. [72].

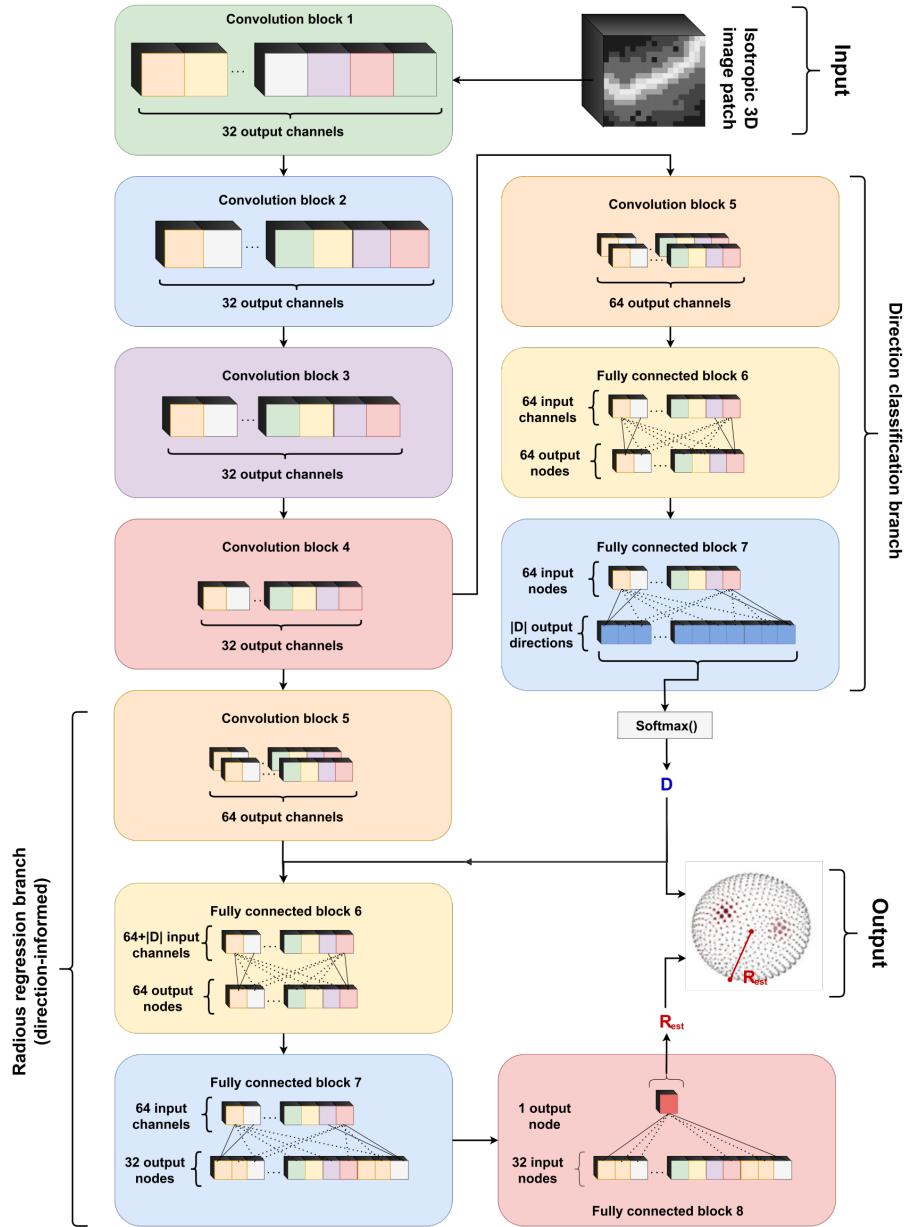


Figure 3.7: Visual representation of the structure of the midsplit-4-FD network. The estimated direction classes are fed into the last layers of the radius regression branch of the network. This network amounts to $\sim 270'350$ trainable parameters.

3.2.1. Training, validation, cross-validation and testing

The standard dataset subdivision for rigorous selection of the best model consists in subdividing the dataset in a training, validation and test disjointed datasets.

The model is initially fit on a *training dataset*, which is a set of examples data used

to fit the parameters of the model. The model is trained on the training dataset using a supervised learning method, using optimization methods such as stochastic gradient descent or Adam optimiser. In the case of the presented CNNs, the inputs are the 3D patches, the outputs are the direction classification and radius prediction, and the target (or label) are the known correct classification of the direction and the reference value of the radius. The model is run with the training dataset and produces a result, which is then compared with the target, for each input in the training dataset. Based on the result of the comparison and the specific learning algorithm being used, the parameters of the model are adjusted. The model fitting can include both input variables and dimension selection and parameter estimation, even if in this case the focus is only on learning the best values for the network's trainable parameters. The goal is to produce a trained model that generalizes well to new, unknown data.

A *validation dataset* is a set of examples data and labels used to tune the hyperparameters and the architecture of a model. An example of hyperparameters for CNNs include the number of hidden layers, the number of filters (output channels) for each layer, kernel size, dilation factor. It, as well as the testing set, should be coherent with the training dataset. In order to avoid overfitting, when any model parameter needs to be adjusted during training, it is necessary to have a validation dataset in addition to the training dataset: whilst training a model, the hyperparameters of the model are updated at each training step, and the evaluation on the training and validation datasets is performed at each full training cycle (also called a "epoch"), after which the training cycle is repeated on the same training data. In this framework, the evaluation on the validation dataset is used to determine the best set of parameters of the network, so that only the set of parameters which enhance performances on the validation set with respect to the previous epoch are selected to be stored as the candidate final set of trained parameters for the trained model.

A *test dataset* is a set of examples data and labels used only to assess the generalization performance of a fully specified classifier: if the most suitable model for the problem is sought, the training dataset is used to train the different candidate classifiers, the validation data set is used to compare their performances and decide which one to take and, finally, the test dataset is used to obtain the performance characteristics such as accuracy, sensitivity, specificity, and problem-specific metrics. To do this, the final model is used to make predictions on examples in the test set. Those predictions are compared to the true reference predictions to assess the accuracy of the model.

In a scenario where both validation and test datasets are used, the test dataset is typically used to assess the final model that is selected during the validation process. In the case

where the original dataset is partitioned into just two subsets (training and validation/test datasets), the test data set might assess the model only once. When using a method such as cross-validation, two partitions can be sufficient and effective since results are averaged after repeated rounds of model training and testing on cycling datasets to help reduce bias and variability of the validation metrics, at the cost of increasing training and testing time.

In the Rotterdam Coronary Artery Evaluation Framework 8 CCTA images are presented with reference centerlines annotations, while for the other 24 images the annotations were kept private by the organisers for testing, and were not distributed to the public. The organisers provided during the last 10 years through their website <https://bigr.nl/> an online tool to submit the centerlines extracted from the 24 CCTA test images, and the tool automatically computed numerous metrics and assigned scores to the submitted methods. The evaluation framework filled in time a competition leader-board containing a ranked list of all the submitted methods so far.

Quite recently, however, the evaluation framework seems to have become unreachable, which makes testing on the test dataset impossible since the reference centerlines were not distributed to the public.

The unavailability of the standardised evaluation framework is an issue for three main reasons.

It is not possible to test and compare the proposed networks and trackers with other state-of-the-art methods in a standardised way: comparison would require either to re-implement the test program in the same exact manner as the (unknown) test algorithm of the Rotterdam evaluation framework, or it would require to re-implement all the current state-of-the-art approaches and compare them locally with a local standardised test framework. Both options are unfeasible.

Moreover, a test framework for the evaluation of the proposed networks and comparison with Wolternik's network [99] re-implementation has to be created locally to test and compare their performance consistently and independently from the metrics used during training/validation (further discussed in chapter 4).

The CAT08 training dataset is not numerous enough to allow a rigorous division in training, validation and test datasets. It is enough to consider that the Rotterdam evaluation framework keeps 75% of the dataset just for testing. Following this proportion would lead to a far insufficiently trained network and no conclusions could be drawn from such a testing approach.

The rigorous evaluations will be performed using the Rotterdam standardised framework when it will be available again. In the meanwhile, the 7-1 dataset subdivision, the strate-

gies explained next and the test framework built locally to test the networks proved to be sufficient to draw conclusions on the improvement potential expressed by the proposed networks.

Another standing issue concerns cross-validation, a resampling method that uses different portions of the data to test and train a model on different iterations, so that at every cross-validation iteration the training and validation sets change with respect to the previous one. The goal of cross-validation is to test the model’s ability to predict new data that was not used in estimating it to give an insight on how the model will generalize to an independent dataset; the objective is similar to the classic validation’s, however here the process is repeated a number of rounds. One round of cross-validation involves partitioning a sample of data into complementary subsets, performing training on one subset, and validation on the other subset. Multiple rounds of cross-validation are performed using different partitions, and the validation results are averaged over the rounds to give an estimate of the model’s predictive performance.

During the evaluation of the proposed models, cross-validation was not implemented for practical reasons: training of the proposed networks takes a lot of time, sometimes up to a day depending on the computational power available on the training device. This would mean that each cross-validation cycle would require 8 days, plus the time of testing all the network’s trained parameter sets and averaging out all the obtained results to build useful metrics. It was then decided to lean towards a more agile prototyping at the loss of a slightly less rigorous testing phase. As it will be discussed in the related chapter, tests between networks architectures were performed by comparing networks trained in the same exact conditions: everything being the same, just the network architecture changed during training and testing. This should even out any overfitting on the validation dataset between the different trained networks, and should make the networks comparable in a fair way.

3.2.2. Wolternik’s training strategy and its re-implementation

Wolternik et al. proposed the following training strategy [99]. To sample a training point, a location x along the reference centerline is randomly selected and a 3D image patch P centered at that point is extracted from the training image. The reference radius R_{ref} at x is taken from the reference standard, and reference directions are determined as follows. First, a point x' is found at a distance R from x along the reference centerline. The displacement vector $\vec{\Delta}$ between x and x' is used to determine a reference direction. For this, direction class $d \in D$ with the smallest angle to $\vec{\Delta}$ is assigned as the reference direction. This direction is considered to be one of two reference directions at point x

and its class probability is set to 0.5 in the reference distribution over D . This process is repeated for the opposite direction of the centerline. Hence, all other direction classes have probability 0.0. The patch P , the reference distribution over D , and the reference radius R_{ref} together form an input sample (input + labels) for the CNN during training. They augmented available reference centerline annotations in two ways. First, samples that are located off the coronary artery centerline were included. If the CNN were to be trained just with samples that are exactly on the coronary artery centerline, the CNN may provide incorrect predictions when the tracker finds itself slightly off the centerline, causing the tracker to deviate without recovery. Like on-centerline samples, off-centerline samples are extracted using the reference annotations. A point x on a reference centerline is randomly sampled. To then obtain an off-centerline sample, point x is translated using a random shift sampled from a 3D normal distribution with $\mu = 0.0$, $\sigma = 0.25r$. To determine the reference direction from x , the closest point on the reference center line is identified and from this point x' is found at a distance R along the centerline. Then the displacement vector $\vec{\Delta}$ and reference direction is determined as previously stated, based on x and x' . This process is repeated for the opposite centerline direction.

The dataset is further enriched by applying random rotations to input patches. Each training patch P is rotated around the x , y , or z axis with a random angle $\phi \in [0, 2\pi]$. This balances the orientation of vessels in the training set and makes the CNN agnostic to the orientation of the image.

During training, the Adam optimizer [40] updates the network parameters θ to minimize the loss

$$l(\theta) = l_{CE}(\theta) + \lambda_r l_{MSE}(\theta) + \lambda_w \|\theta\|^2$$

where l_{CE} is the categorical cross-entropy between the reference and posterior probability distributions over the direction class set D , l_{MSE} is the squared error regression loss between the reference and predicted radius values, weighted by a parameter λ_r , and $\lambda_w \|\theta\|^2$ is a regularization term on the network parameters. To balance the contribution of each term to the loss function, they used $\lambda_r = 10$ and $\lambda_w = 0.001$ throughout the training. Mini-batch training is used with batches containing 64 randomly selected samples. Learning rate decay was used, where the learning rate started at 0.01 and was reduced by a factor 10 every 10'000 iterations, for a total of 50,000 iterations.

Wolternik's training strategy [99] was re-implemented in python using PyTorch data structures, similarly to the original description but with some minor changes. Patch rotations are made probabilistic both on the rotation quantity and on their happening probability, and are applied mainly to the offset data. Offset data are considered in highest number with respect to the non-offset data (15-to-1 proportion) since, during tracking, it is far

more likely to land on a point which is not "precisely" on the centerline.

The punishment factor on the radious loss λ_r is increased from 10 to 15, in an attempt to enhance the radious estimation performance that otherwise results too poor.

The initial learning rate lr_0 is decreased from 0.01 to 0.001 because the global loss appears too fuzzy otherwise, while starting from 0.001 the losses show the classic smooth descending learning behaviour. Also, the main training strategy implements cycles over 400 epochs (150'806 training samples, 24'098 validation samples per epoch), during which the learning rate would decrease exponentially as follows:

$$lr(epoch) = lr_0 \left(1 - \frac{epoch}{400}\right)^{1.5}$$

so that the learning rate is exactly 0 at epoch 400.

Gaussian noise with $\mu = 0.0$, $\sigma = \text{Uniform}(0, 0.1)$ is added to the training input patches with 20% probability.

3.2.3. Further training strategies

The above stated training strategy was used as the basic training strategy to train and compare all the networks: of all the attempted modifications, the ones mentioned above were the ones performing best.

Nonetheless, before getting to this conclusion, many attempts were done in further optimising the training strategy with respect to the original one proposed in the paper.

Initial learning rate setting and the maximum number of epochs were two major testing grounds, and extensive testing resulted in the choice of the values 0.001 and 400, respectively. A higher learning rate meant almost random behaviour especially by the radious validation loss function, while 400 was found to be the limit over which no improvements were made during training.

The possibility to improve radious estimation accuracy was first explored through changing the loss function: training with a L1-L2 radious loss composed of an L1 loss for error values between 0 and 1 and L2 loss for greater errors was attempted, with no concrete success.

Also the punishment factor for the radious loss were tweaked to no success.

Another attempt was made to increase the performance of the basic training strategy. The dataset was further augmented by inserting mirroring on the X, Y and Z for each on-centerline dataset point. The motivation behind this choice is that patch rotations are performed mainly on offset patches which, in the basic implementation, surpass the on-centerline numbers of 15-to-1. Even if the patches were to have all the same probabil-

ity to be rotated, rotations would mostly happen on offset patches. To make up for this disproportion, mirroring along the axis was added to the not-offset patches.

For what concerns the midsplit-4-FD network, it was attempted to pass the estimated direction probabilities to the radious estimation branch without first applying the softmax operator. This resulted in an almost random behaviour of the radious validation loss during the first stages of training. The behaviour was adjusted by performing the softmax operator before back-feeding the direction classes into the radious estimation branch.

3.3. Centerline trackers

The tracking algorithm is at the core of the automatic extraction of the centerlines. In python, it is convenient to build a Tracker class which takes care of all the aspects of the tracking procedure. Through object oriented programming, a basic Tracker class was defined, inside of which any of the proposed networks could be plugged in through an interface. Any other tracker can be created as an extension of the basic Tracker. The tracker needs a neural network to interpret its surrounding and make predictions on what are the viable paths, however the network needs the tracker to "move around" the centerline and hop from the current center of the patch x to the next one through the displacement vector, and to do it in a consistent way. It can be thought as the tracker being a vehicle with sensors and a position-recording system, and the neural network as the autopilot, the brain of the vehicle.

3.3.1. The basic Tracker class

The basic Tracker class contains all the variables and methods which enable the movement and control of the network and the recording of the visited landing points hop-by-hop. The tracking algorithm starts at a seed point x_{seed} . An isotropic 3D patch P_0 centered at x_{seed} is extracted by the Tracker and processed by the CNN. The output of the CNN consists of the posterior probability distribution $p(D|P_0)$ over possible directions, and the estimated radius value R_0 .

To determine two initial opposing directions of the tracker for its first jump, two local maxima d_0 and d'_0 separated by an angle $\geq 90^\circ$ are identified in $p(D|P_0)$. The tracker will first follow the centerline in the d_0 direction until termination, and then in the direction d'_0 until termination.

To follow the centerline in the direction d_0 , the basic Tracker takes a step of specified length towards d_0 and arrives at point x_1 . The step size can depend on many factors and any tracker can define its logic to determine the step size. The basic Tracker takes the

smallest possible step size in order to land in a pixel different from the current center of the patch P . Control on the lower bound of the step size to land on a new pixel is performed by default by the basic Tracker class and it is inherited by all the discussed trackers.

Then, a new patch P_1 is extracted at x_1 and processed by the CNN to provide $p(D|P_1)$ and R_1 . The new tracking direction d_1 is selected as the direction with the highest probability in $p(D|P_1)$. To prevent the tracker from moving backwards, directions that have an angle $\geq 90^\circ$ to d_0 are excluded by default from the selection. This process is repeated until a stopping criterion is fulfilled. Subsequently, the tracker follows the same process in the second initial direction d'_0 , starting again at point x_{seed} .

Termination of the tracker is guided by a stopping criterion. Each tracker can implement its own stopping criterion thanks to the intrinsic modular structure, which can be based on hard constraints or learnt conditions as well. The basic Tracker's stopping criterion is based on the uncertainty of the direction classifier proposed by Wolternik et al. [99]. At each point along the extracted centerline, the normalized entropy $H(p(D|P)) \in [0, 1]$ of the posterior probability distribution is computed as follows:

$$\theta_H = H(p(D|P)) = \frac{\sum_{d \in D} -p(d|P) \log_2 p(d|P)}{\log_2 |D|}$$

The movement terminates if the entropy of the selected probability distribution crosses a threshold value $\theta_H = 0.9$. The basic Tracker also implements a safety hard-stop criterion, which consists in stopping the movement if a maximum number of jumps is performed in a single branch (default value at 300); this is necessary for stopping the tracker in the case it gets stuck into an infinite loop.

The tracker interfaces with a class that saves the path, radii, and termination criterion value for each branch of the movement into a data structure for each seed.

3.3.2. Wolternik's tracker

The basic Tracker was based on Wolternik's iterative tracking algorithm [99]. Instead of creating a linear algorithm, a basic inheritable Tracker class provides the basic building blocks from which any other tracker based on a model inferring the most likely next direction through the probability distribution D (in the considered case a CNN) can be built upon.

Wolternik's tracker introduces two upgrades with respect to the basic Tracker. To follow the centerline in the direction d starting from $x_{current}$, the tracker takes a step of length R_0 towards d and arrives at point x_{next} . This allows the tracker to move faster where the

coronary arteries are wide, and to move in a more cautious way where the arteries offer less room for manoeuvre.

When choosing the direction $d \in D$ to go from $x_{current}$ to x_{next} , the tracker chooses from directions which are less than 60° with respect to the previous jump direction, instead of 90° .

Also the stopping criterion is enhanced. The normalized entropy θ_H of the posterior probability distribution may exceed the hard-threshold when the tracker encounters the coronary ostium or the end of the coronary artery. However, stenotic areas, areas with low image contrast or locations affected by stepping artifacts may also lead to high entropy values. To encourage tracking through such areas, the termination entropy is determined as a moving average over the past three steps.

Wolternik's tracker performs well when enough seeds are planted in the portions of interest of the coronary artery tree, however it performs quite poorly when starting from few seeds (see section 4.2). The limit case is when it starts from just 2 seeds, one planted into the left main coronary artery, the other in the right branch, where the tracker is not able to reconstruct almost half of the coronary artery tree.

Very recent state-of-the-art approaches discussed in section 2.2.2 use additional neural networks to enhance the performance of the tracker. Specifically, a neural network classifier is assigned to the evaluation of the stopping conditions, and a second neural network helped the core CNN interpret the obtained direction distribution D to understand if the vessel presents bifurcations.

These enhancements proved to work really well, however they required additional specifically labeled data to be trained, especially the bifurcation network. In the current conditions each image presents a maximum of just 2 annotated centerlines bifurcations, for a total of 14 bifurcations. This numerosity is far insufficient to teach a neural network to discern from "straight" vessels and bifurcating vessels. A stopping criterion based on a neural network's interpretation of the surrounding is interesting, however seeds placement and tracker termination conditions are not the main objectives of this document, which is instead focused of recreating the entire tracking stack and proposing new CNN architectures enhancing the baseline.

3.3.3. Seedspawner tracker

A possible enhancement to the basic Tracker and the tracker proposed by Wolternik [99] is to add some new functionality that make the tracker more effective in case a low number of seeds (1 per coronary artery branch) are initiated.

Instead of focusing on estimating the entity of a bifurcation through a neural network, a simple solution would be to implement a "seed spawner" which spawns new seeds along the way: new seeds are added iteratively to the original list of seeds, which then grows exponentially.

The seeds spawning is composed of two modules, each adding new seeds to a queue. The first module spawn seeds in a Hansel and Gretel fashion: at each tracker hop, 5 displacement vectors $\vec{\delta}_{seeds}$ are created with coordinates x, y, z drawn from an isotropic uniform distribution [-1.3,1.3] mm. For each drawn $\vec{x}_{NewSeed} = \vec{\delta}_{seeds} + \vec{x}_{PatchCenter}$, if no seed in the original list or in the queue is less than 4 mm away from $\vec{x}_{NewSeed}$, then the seed is added to the queue. The sparse uniform distribution is used instead of gaussian because the main interest of this module is to place seeds nearby the centerline, but far enough to have a chance of falling into new, unexplored side-branches. To compensate for the otherwise exponentially growing number of seeds in the list, this module is subjected to a probability of 10% which decreases to 85% of the previous value at each spawn cycle. The second module tries to place new seeds in the direction of the most likely movement directions, without any constraint on the angles between different directions. From the posterior probability distribution of the directions $p(D|P)$, the top 25% most likely directions are selected. For each direction, with probability $\frac{1}{2}$ a seed $\vec{x}_{NewSeed} = \vec{d} \cdot R_{est} + \vec{x}_{PatchCenter}$ is placed in the queue if no seed in the original list or in the queue is less than 4 mm away from $\vec{x}_{NewSeed}$. The second module depends on the tracker's termination probability: if θ_H is between 0.85 and 0.93, the spawner activates. This makes sure that the spawner does not add useless seeds where tracking is too good, of where it is too difficult which means that the tracker is probably near termination. A value in-between means that the uncertainty on the direction is quite high, but not too much, signalling the presence of an obstacle or a bifurcation, both of which cases can benefit from a seed placed nearby to explore the surrounding in a more robust way.

The seeds are actually added to the original seeds list only if the length of the final tracked path is more than 5 jumps long: this condition helps not placing seeds where the tracker is uncertain of the direction to follow and where the termination criterion is met almost right away in the tracking procedure. A cap of 200 total seeds is imposed, after which no new seed can be added to the list of seeds from which to start exploring.

The stopping criterion is the same as Wolternik's [99], however the threshold angle between subsequent directions was increased from 60° to 90° , as it gives the tracker more freedom to perform sharper turns to explore bifurcations and side branches.

The tracker is still a prototype it is not very robust yet. Moreover, the total time to explore the seeds increases linearly with the number of seeds. However, considering its simplicity

with respect to the neural-network-based bifurcation classifier, it does an excellent job at reconstructing the full tree starting from just two seeds, outperforming Wolternik's tracker [99] in such conditions. Results are compared in section 4.2.

There are two known challenges associated with a tracker approach: one is with its initial seed point and the other is going astray or tracking drift. The second problem is successfully solved by including in the training set also patches centered inside the vessel but not precisely on the centerline. The first problem can be solved in different ways: Wolternik et al. proposed, in the same work from 2018 [99], an extension of their CNN to classify each point in the CCTA image and place seeds completely automatically. The problem is, however, that their method placed seeds also in other tubular structures such as pulmonary arteries. Salahuddin et al. in 2021 used a deformable shape model of the heart to initialize the automatic coronary centerline extraction at the ostium point of both the main coronary arteries [69]. Since automatic seeds placement in the coronary arteries constitutes itself a challenging task to solve that deserves its own space, in this document it is assumed that the seeds are already placed inside the vessels (not necessarily on the centerline).

3.4. Post-processing centerlines graphs

What is obtained from the centerline tracking procedure is an unstructured set of filaments which retain no information on their membership to the left or right coronary arterial branch. No information is retained about what end, if any, is near the coronary ostium, and what end is distal to it; it can also happen, even if quite rarely, that a tracker follows the arteries through two different side branches, thus having two endpoints as extrema. The obtained centerlines can then be considered an unstructured 3D cloud of points, each having coordinates x, y, z ($, R$).

The goal of this post processing procedure is to fully-automatically bring order into the structure, classify each point as part of either the left or right arterial tree, and for each tree to build an oriented graph that, starting from the ostium, connects the whole arterial tree. In this way, each couple of ostium-endpoint is connected and the path between them can be easily accessed through common graph algorithms.

The first step of the post processing procedure is to clean and prune the original cloud of points. Each tracked branch is trimmed of 4 points towards its end, and joined with its other half of the extracted filament (one half was obtained from d_0 , the other half from

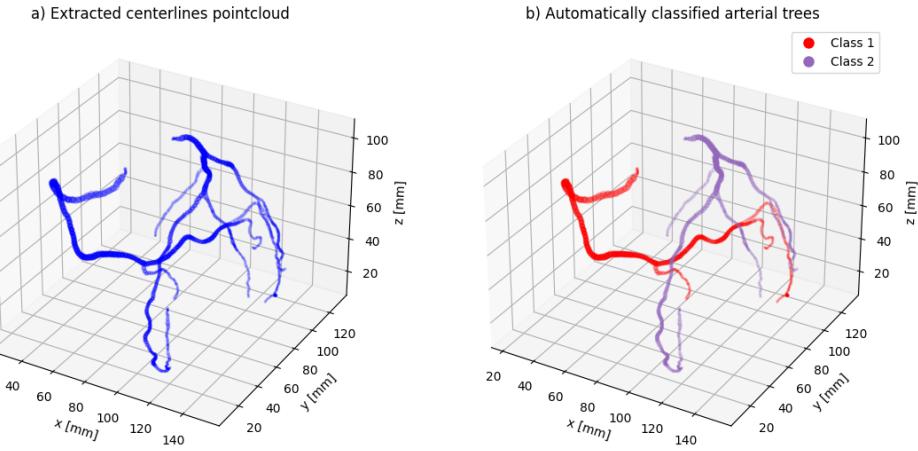


Figure 3.8: a) The original centerlines unstructured point cloud. b) The two arterial trees are automatically classified correctly into two disjointed sets through a DBSCAN-inspired algorithm. To test the algorithm, a collage of 3 extraction process was created so that the resulting arterial trees are almost in contact, thus representing the hardest possible (yet unlikely) condition for the algorithm. The algorithm is able to separate the two arterial trees by considering 6 nearest neighbours.

d'_0 .

All filament shorter than 20 steps are discarded since it was empirically observed that such filaments bring no contribution to the whole tree and more often than not they are just wrongly tracked filaments which did not stop soon enough.

The remaining points are gathered in a unique, unstructured point cloud. All points which appear more than once in the point cloud are pruned until each point is unique in the set. This phenomenon happens because tracking through a neural network is a deterministic process: every time the tracker land on a pixel, the resulting radius and directions estimations would always be the same. Since during tracking it is likely that, sooner or later, the tracker ends up on an already visited pixel, some portions of the tracked centerline result the same. This is another issue that the proposed seeds spawner tracker partially solves by increasing the number of seeds, thus increasing the likelihood of exploring new paths and obtain a more robust final centerline estimation.

Once the centerlines points cloud are pruned, the left and right arterial trees are divided into two disjointed sets by means of a DBSCAN ([23]) inspired algorithm. Instead of comparing points by means of a fixed distance threshold, the algorithm connects the first 6 nearest neighbors of each point in the point cloud. To find the 6 nearest neighbors,

only the shortest 6 euclidean distances between the considered point and all the other points are sought instead of sorting all the distances and selecting the shortest 6; this saves more than half the time. Starting from a random seed point, the algorithm saves all the connected points of the seed in a queue and classifies them as members of the same class as the seed. The procedure repeats for every one of the points in the queue, until no point is left in the queue. The obtained set of points is one arterial tree, while the remaining set of points are part of the other arterial tree (figure 3.8).

It is now possible to work with one arterial tree at a time. The next step is to find all the endpoints in the unstructured cloud of points, which is likely the hardest task in the procedure. The task is particularly hard because the extracted centerline points are not neatly distributed along a mono-dimensional filament, but around a mono-dimensional filament, with noise. Considerations purely derived on the angles do not work in this case, and finding heuristic conditions to classify between an endpoint and a point which should be on the centerline but is a little too noisy does not work. The endpoints are recognised, but also centerline points are recognised as endpoints as well.

A solution is to consider, for each point, 10 - 20 nearest neighbours which are at least 4 mm away (and no more than 10 mm away) from the considered point. If the versors towards the nearest neighbours are detected – again, through the DBSCAN approach – to be clustered on more than one 3D direction, then the point is dismissed; otherwise, the point is classified as a candidate endpoint. An example of versors clustered into two distinct groups for a noisy centerline point is shown in figure 3.9. This produced many endpoints near the final/initial portion of the vessel, so any candidate point which is in a 4 mm range from another candidate endpoint gets averaged, to find a unique endpoint. This solution is not robust and has to be properly tested in many scenarios yet, however a first prototype managed to correctly classify almost all the endpoints in the point cloud (figure 3.10).

Once the endpoints are found, the ostium is classified as the point having the highest maximum radius among its 30 nearest neighbors. This procedure, together with the pruning performed in a precedent step, is quite effective as the ostium is the portion of centerline having the maximum radius.

Then, all the points are transformed into a graph (one point is one node) and connected to their 6 nearest neighbours. The connection weight is defined as the euclidean distance between the two connected nodes. The shortest path between the ostium point and each one of the endpoints is found through the A* algorithm [30]. The shortest patch passes mostly to the center of the filament point cloud, however when hard turns are presented it drifts a little towards the minimum-length side of the curve.

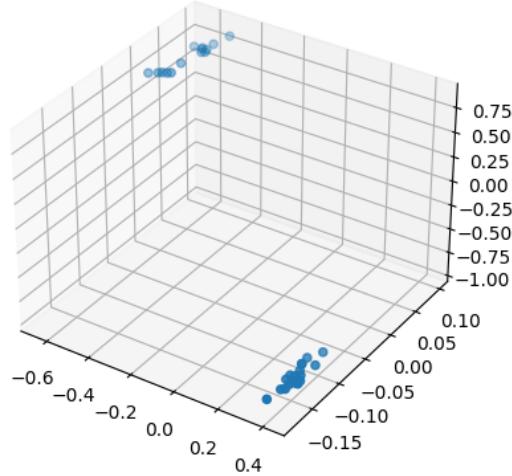


Figure 3.9: Clustered versors in the 3D space of a point in the centerline. It is clear that the versors point towards two directions, which means that this point is not eligible to be a candidate endpoint.

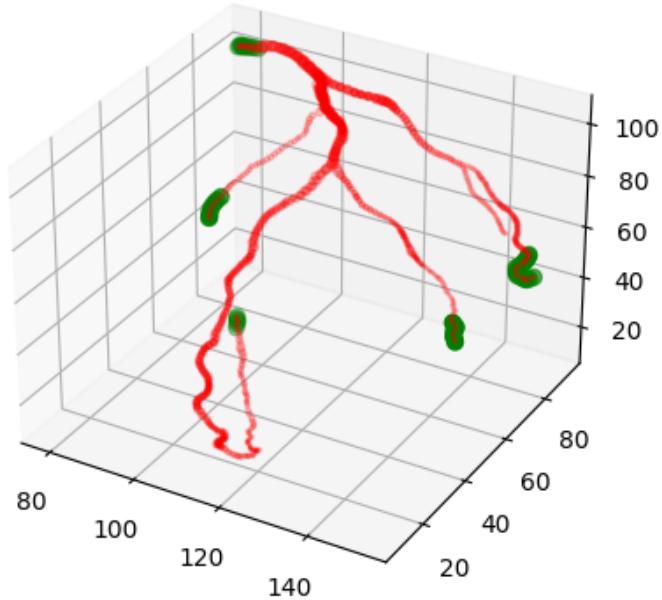


Figure 3.10: Green endpoints found by the proposed procedures. Many points are found near the extremis, it is sufficient to classify them once again (this time the task is far easier) and taking the mean for each cluster.

Starting from the ostia, a bi-directed graph is created from all of the points on the shortest path, and saved. Since the nodes are saved into a graph object through a list, indexing is hierarchical which means that accessing the node list from the first element to the last is equivalent to moving along the centerline from the coronary ostium to the endpoint. Subsequent extracted paths can be combined easily together by merging the nodes in common between two paths up until the bifurcation happens, at which point one node will be connected to two other nodes.

As for now, the algorithm is still in the testing phase and it does not work very robustly, especially the endpoints detection algorithm. However it has the potential to not only offer a ordered representation of the centerline tree, but also to improve centerline extraction accuracy. In fact, assuming that the extracted points are distributed normally around the reference centerline (assumption which is confirmed in section 4.1.1 by looking at the images) the path passing through them should be a better estimation of the actual centerline.

In the next chapter two test frameworks for the evaluation of both the neural networks and the trackers are discussed and the results presented.

4 | Test suite and results

In this chapter, the two test suites created to asses the quality of the trackers and the CNNs at their core are described. For each suite, the results obtained from tests of the most promising networks and trackers are discussed.

The creation of two evaluation frameworks were necessary for two reasons: the main reason is that, during fast prototyping of the networks and trackers, a quick way to asses their quality is needed. The second reason is that the Rotterdam standard evaluation framework, which should be reachable from the Biomedical Imaging Group Rotterdam official website <https://bigr.nl/>, seems to be offline at present day.

Two evaluation frameworks are presented.

The local test suite is used for assessing the quality of the trained neural networks in a quantitative way, while the global test suite is used for assessing the quality of the centerline tracking and reconstruction mostly, but not entirely, in a qualitative way.

4.1. Local test suite for CNN assessment

The *local test suite* is used to obtain standardised metrics to compare different training strategies for the same network (different learning rate, data augmentation, loss functions) and, mainly, to test the new networks' architectures against Wolternik's [99] and against each other. When testing networks architectures against each other, the networks were trained all in the same conditions, discussed in section 3.2.2.

The local tests are performed in a completely standardised way, so that every network is tested in the same exact conditions. The local test program is divided in three steps.

First, a standard set of test datapoints is generated around the centerlines in a procedural way, where each test point is unique in the set and corresponds to the center of a unique image patch to be tested. For each datapoint, the following information are stored:

- X, Y, Z indexes of the central pixel of the $19 \times 19 \times 19$ input patch.
- 3D coordinates of the central pixel of the $19 \times 19 \times 19$ input patch.

- Coordinates (x,y,z,R) of the reference centerline point closest to the central pixel.
- Euclidean distance D between the central pixel and its closest point (redundant, but handy).
- Boolean classification between points near the ostia and points at least $1.1R$ mm away from the ostia.
- Reference directions $d_{0,ref}$, $d'_{0,ref}$, created as described in the training dataset description section 3.2.2.
- The actual 3D points on the annotated centerline used to derive $d_{0,ref}$ and $d'_{0,ref}$.

This set of information allows full characterisation of the behaviour of the network through the analysis of correlations between different factors such as, for example, how does the radious estimation error depend on the distance between the center of the patch and its closest centerline point, and many other discussed in the next pages.

The second stage of the test is to let the CNNs perform predictions about the directions and radious by using as inputs the patches centered on the set of test points. To do so, the neural network is "placed" on each test point, and the two directions d_0 and d'_0 as well as the estimated radious R_{est} are obtained by the CNN. From this stage of the testing strategy, 4 pieces of information are saved and stored for every test datapoint: the estimated radious, the two estimated directions (as 3D versors), and the termination criterion numerical value. This last piece of information is computed in a standard way for every tested networks as the normalized entropy $H(p(D|P))$ of the posterior probability distribution on the direction classes, as specified in the basic Tracker (section 3.3.1).

The last stage of the local test performs a fully automated quantitative analysis of the extracted data to compute, store and visualise meaningful metrics. The following metrics are computed:

- Mean and standard deviation of the radious estimation error:

$$R_{error} = R_{est} - R_{ref}.$$
- Mean, standard deviation of and maximum radious absolute error:

$$R_{error,abs} = \text{abs}(R_{est} - R_{ref}).$$
- Mean, standard deviation of and maximum radious relative error:

$$R_{error,rel} = \text{abs}(R_{est} - R_{ref})/R_{ref}.$$
- Mean and standard deviation of the angular error between the estimated and reference direction. The angular error measures the angle between two 3D versors, the reference versors $d_{0,ref}$, $d'_{0,ref}$ and the estimated direction versors $d_{0,est}$, $d'_{0,est}$. To

compute the metric, versors are coupled in such a way that the sum of the two error is minimised (each estimated versors is compared to its intended reference direction versor).

- Mean, standard deviation of and maximum distance from each estimated landing point $P_{landing} = \vec{d}_{0,est} \cdot R_{est} + P_{patch\ center}^{3D}$ and its nearest centerline point $P_{centerline}$. Of all the computed metrics in the local test suite, this is the most significant one for assessing the ability of the network of correctly tracking the centerline. Minimising this metric means minimising tracking errors.
- Mean, standard deviation of and maximum distance from each estimated landing point $P_{landing}$ and its reference point of the centerline on which the network was trained on, $P_{centerline,opt}$. Of all the computed metrics in the local test suite, this is the most significant one for assessing the ability of the trained network of generalising the training. This measure is not fundamental for tracking, however it is the most important metric if the neural network were to be considered out of its working context.

Each metric is computed with respect with each of the following subsets of datapoints:

- Global: the metrics are computed for every tested datapoint.
- Just near ostia: the metrics are computed just for the datapoints that are less than $1.1R$ mm away from the two ostia of the arteries.
- Away from ostia: the complement set of "just near ostia".
- Away from ostia, $D \leq 0.5$ mm: datapoints in the "away from ostia" set for which the center of the image patch is less than 0.5 mm away from the closest reference centerline point. The same analysis is performed for the sets:
 1. $0.5 \leq D \leq 1.0$ mm.
 2. $1.0 \leq D \leq 1.5$ mm.
 3. $1.5 \leq D \leq 2.0$ mm.
 4. $D \geq 2.0$ mm.

This subdivision gives a complete overview of the characteristics and quality of the tested network.

The test suite also builds 6 comprehensive infographics for the qualitative assessment of the performance of each network.

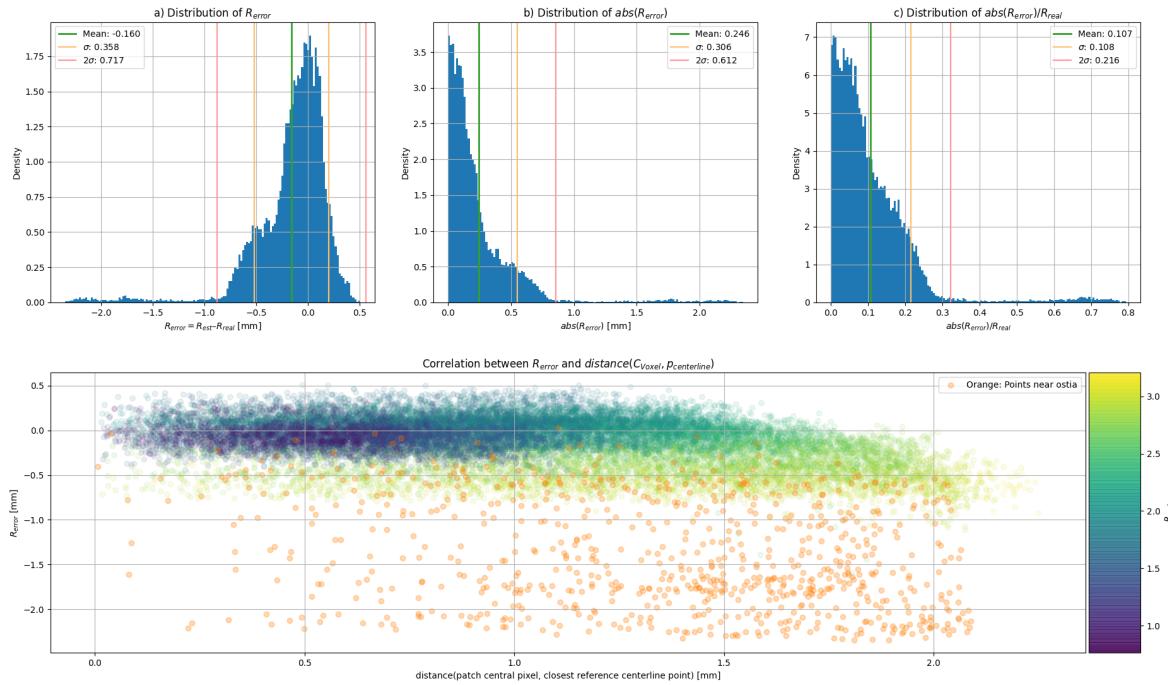


Figure 4.1: Wolternik’s network, standard training strategy. Distributions of R_{err} (a), $R_{err,abs}$ (b) and $R_{err,rel}$ (c). Lowermost: correlation between R_{err} and D. For every infographic shown in this section extensive description of the graphs is reported in the main text.

4.1.1. Local tests results discussion

The local test suite is run for every tested network and training strategy. Following are reported the results of the evaluation for the test performed on Wolternik’s network [99] trained in the standard training conditions described in section 3.2.2. This serves both for showing the information gained with the local test suite and as a way of setting the benchmark that the new proposed networks should surpass. While discussing the results of other networks or training strategies, not all the infographics will be shown or the whole table of results reported, as this would occupy many pages and would result in an unclear discussion. However, it has to be known that the evaluation process was performed also considering qualitative characteristics shown in the infographics.

Results: Wolternik’s network trained with standard training strategy.

In figure 4.1 the first infographic is shown. This image contains information about the distributions of R_{err} (4.1a), $R_{err,abs}$ (4.1b) and $R_{err,rel}$ (4.1c) computed for all datapoints. Histogram 4.1a shows that the distribution is centered in 0 as expected, but the network has a tendency to underestimate the radius.

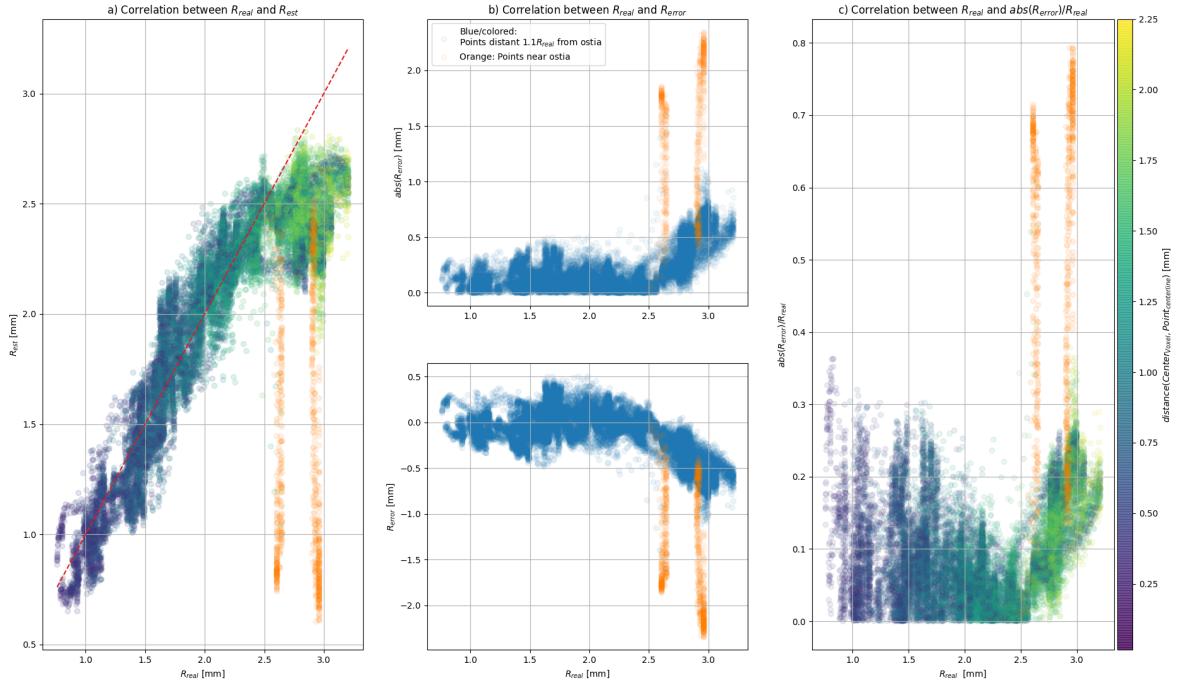


Figure 4.2: Wolternik's network [99], standard training strategy. Dependence of a) R_{est} , b) R_{err} and $R_{\text{err},\text{abs}}$, c) $R_{\text{err},\text{rel}}$ on R_{ref} .

In the image, the lowermost graph shows the correlation between R_{err} and D, the distance between the centerline and the center of the patch. The color gradient indicates the real value of the radius, while orange points are the "just near ostia" subset. This graph shows that the tendency to underestimate the reference radius gets stronger as the patch is centered further away from the centerline. It also shows that, near the coronary ostia, radius estimations get poorer. Poor estimation quality near the coronary ostia is a constant among all the performed analysis results, so it will not be pointed out again.

In figure 4.2 the second infographic is shown. It shows the dependence of 4.2a) R_{est} , 4.2b) R_{err} and $R_{\text{err},\text{abs}}$, 4.2c) $R_{\text{err},\text{rel}}$ on R_{ref} , on the vicinity to ostias (orange), and on D (color gradients). It is interesting to observe as all points near the ostias are distributed on two vertical lines, showing that the network has no clue on how to estimate the radius near the ostias. It is also interesting to notice that R_{err} seems to be independent from R_{real} (R_{ref}) up to 2.25 mm, after which it is underestimated proportionally to R_{ref} .

In graph 4.2c) it is shown that there exist no tangible relationship between the normalized absolute error of the radius and the distance between centerline and center of the patch. If it were, colors should form a gradient going from low to high in the graph, not left to right as instead is shown.

In figure 4.3 the third infographic is shown. It shows the distribution of the angular error

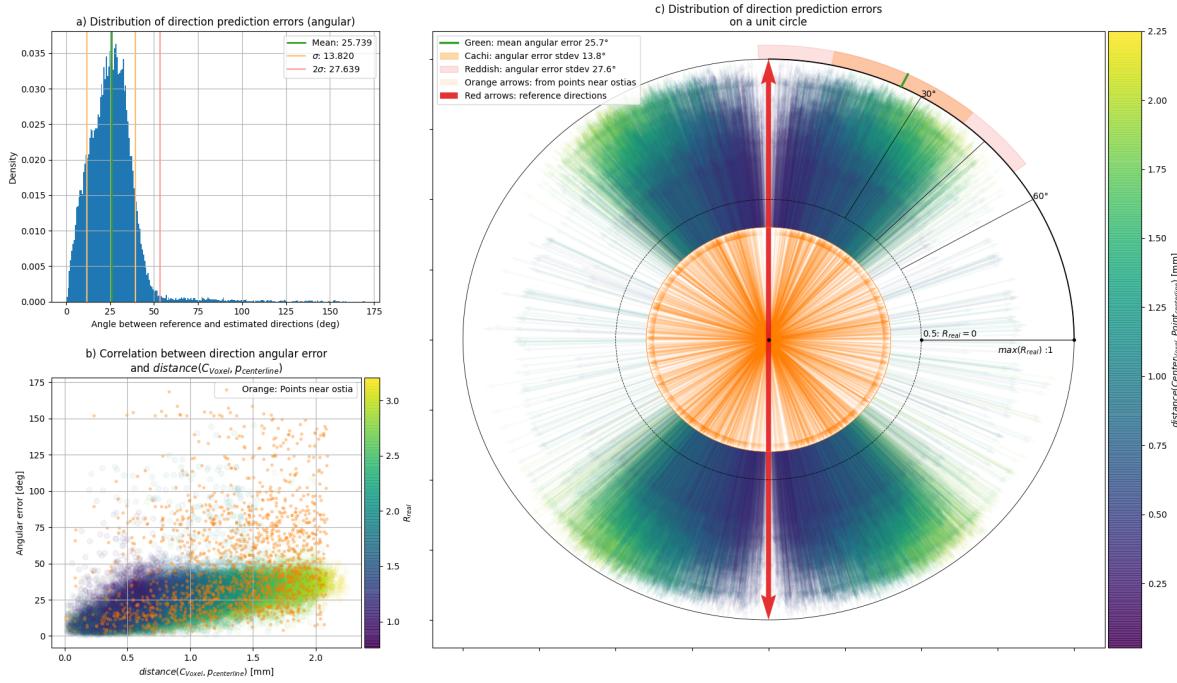


Figure 4.3: Wolternik’s network [99], standard training strategy. Distribution and correlations of the angular error on the estimated and reference directions. The contents of this image are discussed in the main text.

between the predicted directions and the reference directions on which the network was trained on. 4.3a) shows the angular errors distributions on an histogram, while 4.3c) shows them on a unit circle for a more meaningful representation. The two red arrows represent $d_{0,\text{ref}}$ and $d'_{0,\text{ref}}$, while the color-gradient arrows d_0 and d'_0 . The color of the color-gradient arrows represents their dependence on D, while their length represent their dependence on R_{real} . The orange arrows distributed quite randomly on the unit circle are part of the "just near ostia" subset. In the upper-right quadrant of the unit circle outer circumference are shown through curved color bands the global mean, standard deviation and 2-standard deviation of the angular error.

Graph 4.3b) shows that there is no dependency between the angular error and R_{ref} , while there exist a dependency between the angular error and the distance D. This slight correlation is visible also in the color gradient of the arrows on the unit circle distribution.

In figure 4.4 the fourth infographic is shown. It shows how the distance between P_{landing} and the centerline is distributed and on what it is more correlated. From the images is can be concluded that it depends mostly on the distance D, followed by R_{err} (4.4c, color gradient) and the angular error (4.4b, x axis). The global distribution of the distances (4.4a) is an important metric to asses the capability of a network to closely track a

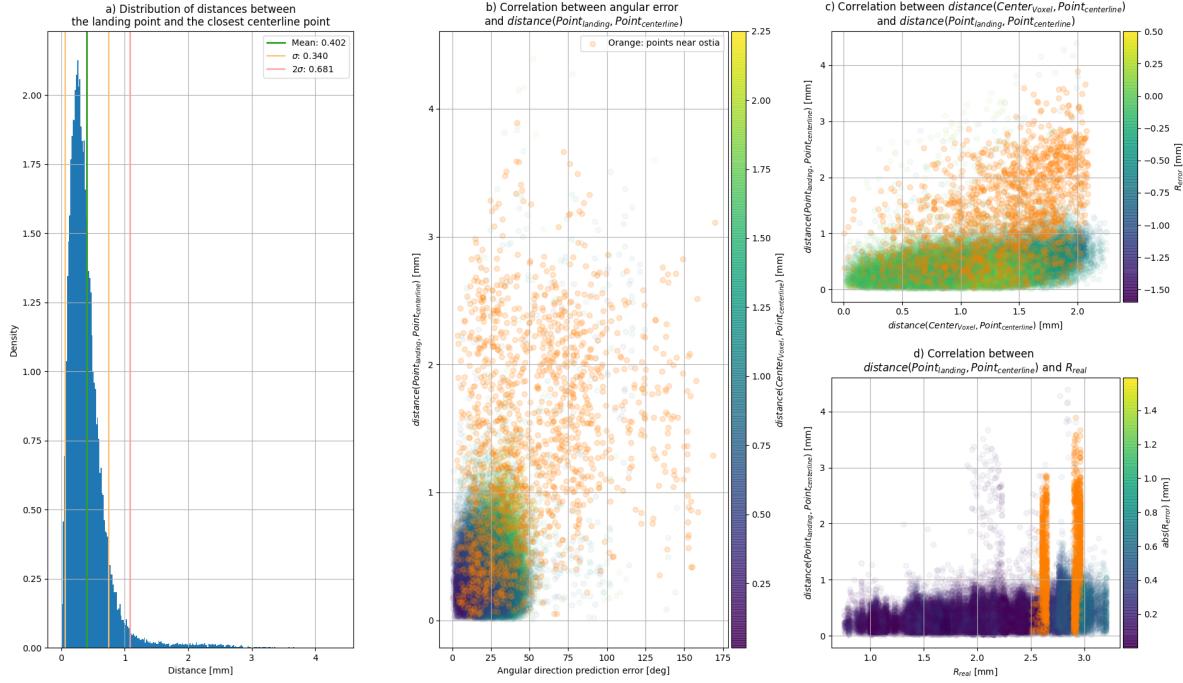


Figure 4.4: Wolternik's network [99], standard training strategy. Distribution and correlations of the distance between $P_{landing}$ and the centerline.

centerline.

In figure 4.5 the fifth infographic is shown. It shows the distribution and correlations of the distance between $P_{landing}$ and the intended landing point on the centerline the network was trained to find, which constitutes the "label" during training by means of the probability distribution D and radius R . This metric is the most important one in assessing how well a neural network performs on the task it was trained on, out of the context of centerline tracking. Graph 4.5c) shows that there is a clear dependency, as expected, between R_{error} and the quality of the prediction of the network; it also seems that this dependency is far stronger than the dependency on the angular error on the estimated direction. This observation offers a clear cue about what kind of performance to focus on while defining new networks structures.

Lastly, in figure 4.6 the sixth infographic is shown. It shows the distribution and correlation of the normalized entropy $H(p(D|P))$ of the posterior probability distribution on the direction classes, which is a metric to derive how certain the CNN is about the directions it predicted. It is evident that there exist a distinction between the "just near ostia" subset (orange) and all others datapoints (blue), however the distinction is not as clear as it would be optimal to be. As a matter of fact, by observing image 4.6b it is clear that the threshold between the two categories is not constant with respect to the

4 | Test suite and results

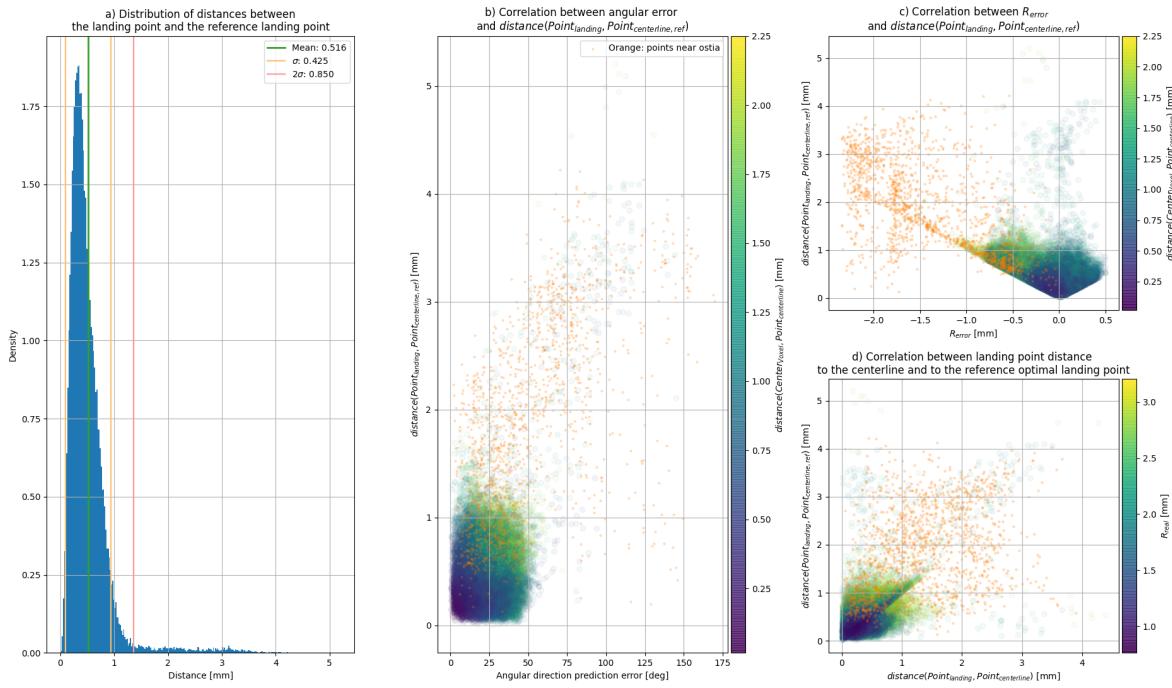


Figure 4.5: Wolternik’s network [99], standard training strategy. Distribution and correlations of the distance between $P_{landing}$ and the intended (reference) landing point on the centerline the network was trained to find (cfr section 3.2.2).

distance between the centerline and the center of the patch, which poses a problem in understanding when the tracker should stop. The distribution in 4.6a also gives an idea on how certain a network is of its predictions, where a lower value of entropy is associate with higher certainty.

While through the infographics a qualitative assessment of the behaviour of the networks is effective, to compare the Wolternik’s network [99] and the proposed networks quantitative data are used.

The local test suite automatically builds a table to assess the quality of the tested network (table 4.1).

The table of results of the local tests will not be reported for every tested network. Results will be discussed in a more agile way in the *Results discussion* section (4.3), and the results will be shown fully just for the best-performing network.

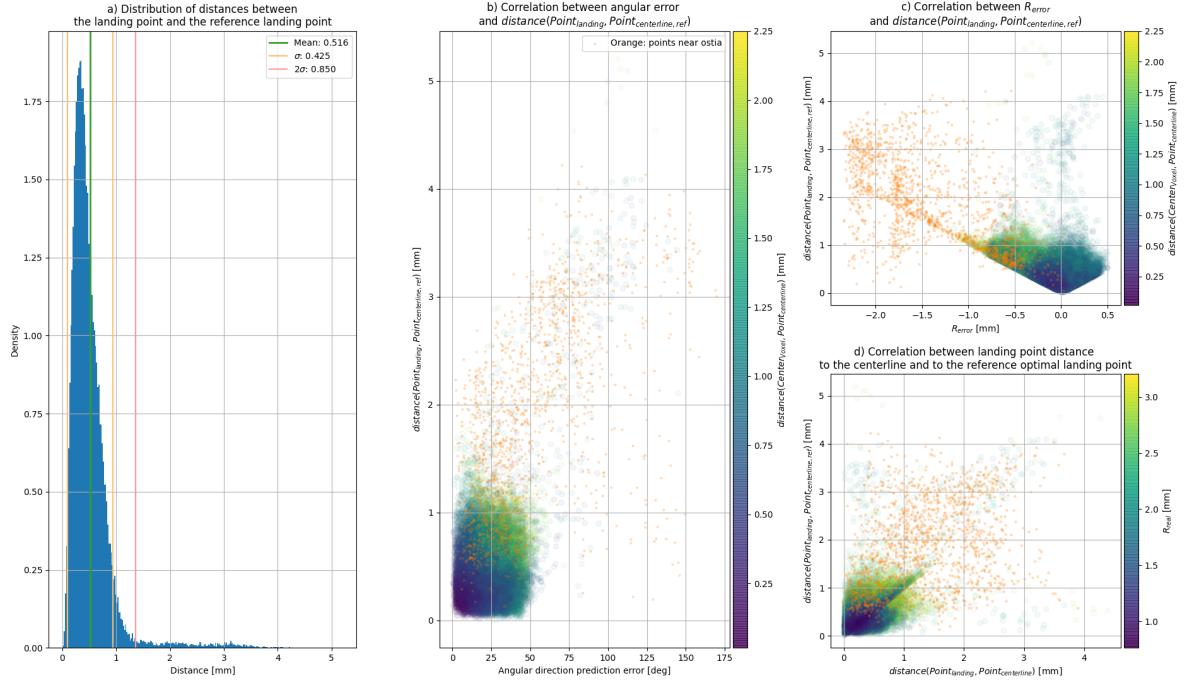


Figure 4.6: Wolternik’s network [99], standard training strategy. Distribution and correlations of the movement termination metric computed through the normalized entropy $H(p(D|P))$ of the posterior probability distribution on the direction classes.

4.2. Global test suite for tracker assessment

The *global test suite* is used to obtain standardised metrics to compare different trackers. Comparisons are performed mainly on the basis of qualitative considerations by graphing the reconstructed centerlines.

The first step consist in reconstructing the centerlines with the selected tracker. To have fair competition, all trackers were tested with the same core network (Wolternik’s [99]) when competing against each other. A fixed set of seeds are extracted from the centerlines, and noise is added to emulate the imperfect placement of automatic seeds. For each seed the tracker follows the vessels and the followed path is saved. Test were preliminary performed with 50 seeds to fine-tune the trackers, then they were tested in two situations constituting the real challenge: one test with 5 initial seeds (at least one in each annotated arterial branching), one test with 2 initial seeds, one per main coronary artery (Left main and Right), shown in figure 4.7.

In the absence of the Rotterdam standard evaluation framework, some metrics are computed taking inspirations from the metrics derived in the official framework, such as the Overlap (OV), Overlap until first error (OF), Overlap with the clinically relevant part of

Subset	GLOBAL (G)	JNO	AO	AO D<0.5	AO 0.5<D<1	AO 1<D< 1.5	AO 1.5<D<2	AO D>2
mean(R_{err})	-0.160	-1.388	-0.114	-0.053	-0.125	-0.180	-0.345	-0.598
mean($R_{err,abs}$)	0.246	1.388	0.203	0.156	0.212	0.249	0.354	0.598
mean($R_{err,rel}$)	0.107	0.494	0.092	0.093	0.092	0.095	0.124	0.198
std(R_{err})	0.358	0.606	0.248	0.203	0.254	0.270	0.239	0.175
std($R_{err,abs}$)	0.306	0.605	0.183	0.141	0.188	0.207	0.225	0.175
std($R_{err,rel}$)	0.108	0.209	0.069	0.071	0.068	0.070	0.074	0.059
max($R_{err,abs}$)	2.345	2.345	1.594	0.764	1.594	1.594	1.594	1.113
max($R_{err,rel}$)	0.793	0.793	0.605	0.363	0.605	0.605	0.605	0.379
mean(α_{err})	25.739	50.586	24.818	13.645	26.843	30.475	33.703	37.061
std(α_{err})	13.820	34.333	11.431	9.509	10.547	9.423	8.152	6.183
mean($d(P_l, P_c)$)	0.402	1.325	0.368	0.274	0.385	0.438	0.544	0.709
std($d(P_l, P_c)$)	0.340	0.779	0.255	0.181	0.263	0.288	0.304	0.263
max($d(P_l, P_c)$)	4.388	3.891	4.388	2.627	4.388	4.388	4.388	3.800
mean($d(P_l, P_{c,opt})$)	0.516	1.848	0.466	0.355	0.487	0.551	0.645	0.806
std($d(P_l, P_{c,opt})$)	0.425	0.898	0.299	0.231	0.305	0.328	0.312	0.243
max($d(P_l, P_{c,opt})$)	5.201	4.221	5.201	3.912	5.201	5.201	5.201	3.904

Table 4.1: Results of the local test on Wolternik's network [99], standard training strategy.

JNO: "just near ostia". AO: "away from ostia". α_{err} : angular error of the estimated directions. $d(P_l, P_c)$: distance between the landing point and the closest centerline point. $d(P_l, P_{c,opt})$: distance between the landing point and the point on the centerline the network was trained to land on.

Reference centerlines and the global test seeds
5 seeds used for global test 2 seeds used for global test

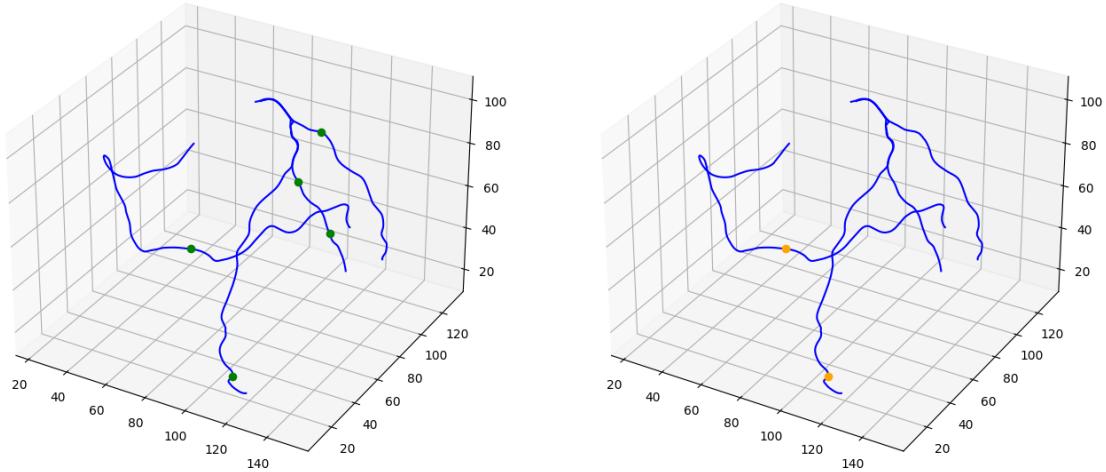


Figure 4.7: Seeds used in the global test suite (green and orange) against the reference centerlines (blue).

the vessel (OT) and Average inside (AI) [72]. These metrics are not implemented in the most rigorous way, they are meant to give numeric validation to the qualitative analysis of the extracted centerlines. Implementing the rigorous metrics would mean to re-implement

the whole evaluation framework which is per-se a challenging task and, even so, my implementation and the official implementation would for sure differ in some way, thus making the obtained results incomparable with the official leaderboard metrics, in any case.

Qualitative assessment of the quality of the tracker is performed by plotting the reference centerlines against the extracted centerlines as well as, for each reference point, the closest extracted centerline point. The number of tracked vessels and the overall quality of tracking are easily observed.

Quantitative assessment is based on four metrics:

- Overlap (OV) represents the ability to track the complete vessel annotated by the human observers and is defined in this test suite as the fraction of annotated centerlines points connected to an extracted centerline point for which the connection is less than R_{ref} long.
- Overlap until first error (OF) determines how much of a coronary artery has been extracted before making an error. An error is defined as having a connection length of more than R_{ref} .
- Overlap with the clinically relevant part of the vessel (OT) gives an indication of how well the method is able to track the section of the vessel that is assumed to be clinically relevant. Vessel segments with a diameter of 1.5 mm or larger, or vessel segments that are distally from segments with a diameter of 1.5 mm or larger are assumed to be clinically relevant [72].
- Average inside (AI) is the average distance of all the connections between the reference standard and the automatic centerline given that the connections have a length smaller than the annotated radius at the connected reference point. The measure represents the accuracy of centerline extraction, provided that the evaluated centerline is inside the vessel.

Metrics are computed on a per-vessel basis. Each reference centerline point is connected to its nearest extracted centerline point. This definition differs from the one proposed in the Rotterdam evaluation framework presentation paper, however it still enable to retain useful information about the centerlines' tracking quality, especially in the case when a little number of seeds is initialised.

Results obtained from testing and comparing different trackers using the global test suite are discussed in the next session.

4.3. Results discussion

Training strategies evaluation

The possibility to improve radious estimation accuracy was first explored through changing the loss function during training: training with a L1-L2 radious loss composed of an L1 loss for error values between 0 and 1 and L2 loss for greater errors was attempted, with no concrete success. While the radious estimation actually improved by 17%, the $d(P_l, P_c)$ metric tested 0.402 mm for the Wolternik-inspired training strategy 3.2.2 while tested 0.89 mm for the L1L2-trained network, which means that its tracking capability actually decreased a lot.

Also the punishment factor for the radious loss was tweaked to no success: a punishment factor increased to 25 yielded an enhancement of radious estimation capability of 12% while decreased the actual landing point accuracy: $d(P_l, P_{c,opt})$ went from 0.402 to 0.417 mm.

The data augmentation strategy by means of mirrored patches (section 3.2.3) resulted in an improved radious estimation and direction estimation, however the $d(P_l, P_{c,opt})$ metric decreased in performance as it went from 0.402 mm to 0.426 mm.

It was concluded that the training strategy inspired from Wolternik's [99] with the adjustments proposed in 3.2.2 was the best option to train the networks and evaluate the proposed network's capabilities.

Proposed networks' evaluation

The local tests performed on the faithful re-implementation of Wolternik's network [99] showed a mean radious estimation absolute error mean($R_{err,abs}$) of 0.246 mm, a mean direction estimation error of 25.7°, and most important of all a mean distance between the network's predicted landing point $P_{landing}$ and the closest centerline point $P_{centerline}$ of 0.402 mm. The test cycle time was 110 s.

On the other hand, local tests performed on the input-layer-split network trained and tested in the same exact conditions of Wolternik's showed a mean($R_{err,abs}$) of 0.197 mm, a mean direction estimation error of 25.18°, and a mean distance between $P_{landing}$ and $P_{centerline}$ of 0.38 mm, which means an increase in performances of circa 19.9%, 2% and 5%, respectively. Also all the other analysed metrics improved, such as the standard deviations of the reported metrics, which are index of a more consistent behaviour. This network performed 50% slower than Wolternik's [99], for a total test cycle time of 168 s. Local tests performed on the midsplit-4 network trained in the same conditions of Wolternik's [99] and input-layer-split showed a mean($R_{err,abs}$) of 0.158 mm, a mean direction estimation error of 24.0°, a mean distance between $P_{landing}$ and $P_{centerline}$ of 0.379 mm, and

an overall better consistency of the predictions which means performance is once more increased with respect to both Wolternik's [99] and the input-layer-split network. Also all the other analysed metrics improved with respect to both networks, which shows that Wolternik's network [99] was indeed undersized and that sharing the first layers not only reduces the size of the network by 33% with respect to the double network, but it also increases its performance. Most noticeably, this network is capable of better predicting the correct $P_{landing}$ with respect with the optimal landing points it was trained on, showing a performance increase of 11.8% and 3.6% with respect to Wolternik's [99] and input-layer-split networks, respectively. The network also reduced computation times by 25% with respect with its heavier predecessor, with a test cycle time of 126 s.

Local tests performed on the midsplit-4-FD network trained in the same conditions of the previously discussed networks showed a mean($R_{err,abs}$) of just 0.14 mm, a mean direction estimation error of 25.26°, a mean distance between $P_{landing}$ and $P_{centerline}$ of 0.35 mm, and an overall better consistency of the predictions (lower standard deviation for each error) which means another increase of general performance with respect to the previous networks. This network can predicting the correct $P_{landing}$ with respect with the optimal landing points it was trained on better than the midsplit-4 network, showing a performance increase of 17.6% and 6.6% with respect to Wolternik's [99] and midsplit-4 networks, respectively. It performed the test cycle in 133 s.

Table 4.2 proposes a general overview of the 4 discussed networks with their characteristics and benchmark test results.

CNN name	Domain	mean($R_{err,abs}$) [mm]	mean(α_{err}) [deg]	mean($d(P_l, P_c)$) [mm]	mean($d(P_l, P_{c,opt})$) [mm]	Learnable params	Test exec. time [s]
Wolternik's	G	0.246	25.74	0.402	0.515	176'533	110
	AO	0.203	24.81	0.368	0.466		
Input-layer-split	G	0.197	25.18	0.380	0.472	353'066	168
	AO	0.152	24.56	0.346	0.422		
Midsplit-4	G	0.158	24.04	0.379	0.454	236'309	123
	AO	0.114	23.70	0.347	0.419		
Midsplit-4-FD	G	0.1399	25.26	0.3498	0.424	270'357	133
	AO	0.097	24.58	0.313	0.374		

Table 4.2: Table showing the most relevant metrics for evaluating the tested CNNs. The meaning of the symbols are explained in section 4.1 and in table 4.1.

Of all the metrics, the most relevant ones to understand how well the network is trained are $\bar{R}_{err,abs}$, $\bar{\alpha}_{D,err}$ and $|\vec{P}_{landing} - \vec{P}_{training}|$, while the most relevant to understand how well the network should perform during the actual tracking task is $|\vec{P}_{landing} - \vec{P}_{centerline}|$: the lowest this metric, the better the network should be able to track the centerline, on average. These and other metrics obtained from the tests on the networks (called "local tests" in this document) are discussed in details in sections 4.1.

Subset	GLOBAL (G)	JNO	AO	AO D<0.5	AO 0.5<D<1	AO 1<D<1.5	AO 1.5<D<2	AO D>2
mean(R_{err})	-0.069	-1.301	-0.023	0.003	-0.028	-0.052	-0.148	-0.326
mean($R_{err,abs}$)	0.140	1.301	0.097	0.057	0.104	0.127	0.170	0.326
mean($R_{err,rel}$)	0.061	0.466	0.046	0.038	0.048	0.051	0.060	0.107
std(R_{err})	0.294	0.614	0.131	0.079	0.138	0.158	0.155	0.113
std($R_{err,abs}$)	0.267	0.614	0.091	0.054	0.094	0.108	0.131	0.113
std($R_{err,rel}$)	0.096	0.218	0.039	0.036	0.040	0.041	0.044	0.036
max($R_{err,abs}$)	2.283	2.283	1.500	0.436	1.500	1.500	1.500	0.743
max($R_{err,rel}$)	0.772	0.772	0.568	0.239	0.568	0.568	0.568	0.243
mean(α_{err})	25.261	43.585	24.582	10.912	27.060	31.388	35.177	39.444
std(α_{err})	12.919	29.123	11.348	7.435	10.104	8.666	7.251	7.727
mean($d(P_l, P_c)$)	0.350	1.324	0.314	0.229	0.329	0.368	0.433	0.527
std($d(P_l, P_c)$)	0.316	0.794	0.208	0.143	0.215	0.238	0.254	0.319
max($d(P_l, P_c)$)	4.438	4.438	4.313	2.397	4.313	4.313	4.313	4.313
mean($d(P_l, P_{c,opt})$)	0.425	1.794	0.374	0.256	0.396	0.450	0.506	0.605
std($d(P_l, P_{c,opt})$)	0.398	0.870	0.254	0.171	0.260	0.284	0.265	0.337
max($d(P_l, P_{c,opt})$)	5.526	5.218	5.526	3.658	5.526	5.526	5.526	5.526

Table 4.3: Results of the local test on the proposed midsplit-4-FD network, standard training strategy. These results can be directly compared to the ones reported in table 4.1. Metrics are enhanced both globally and in AO conditions.

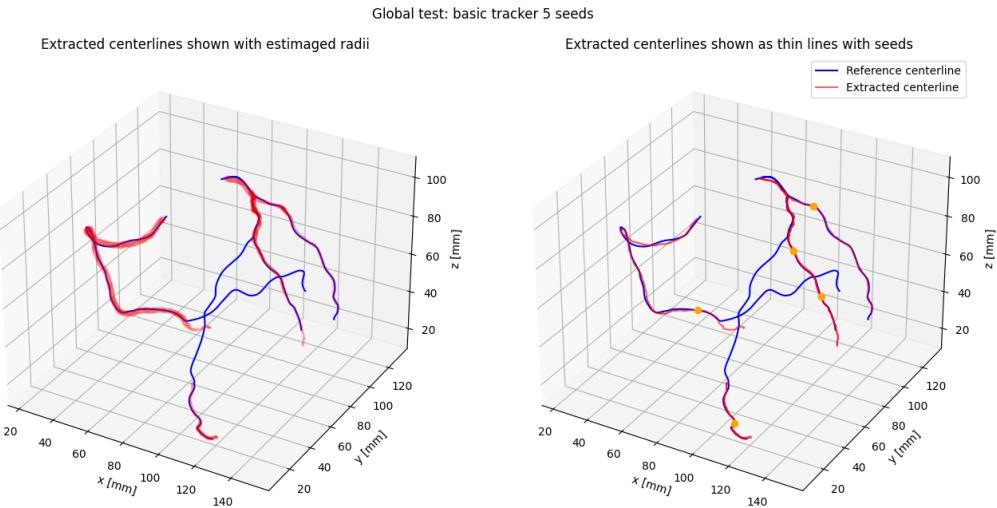


Figure 4.8: Centerlines extracted from 5 seeds with the basic tracker.

Trackers evaluation

Three trackers competed against each other in the global test suite to extract centerlines starting from 5 seeds and from 2 seeds. All trackers were equipped with the same neural network (Wolternik's [99]) for fair competition and started tracking the centerlines from the same exact 5 and 2 seeds, shown previously in figure 4.7.

The first tracker to be tested is the basic tracker. Qualitative analysis is enough to show that, even starting from 5 seeds, its tracking performance is far insufficient (figure 4.8).

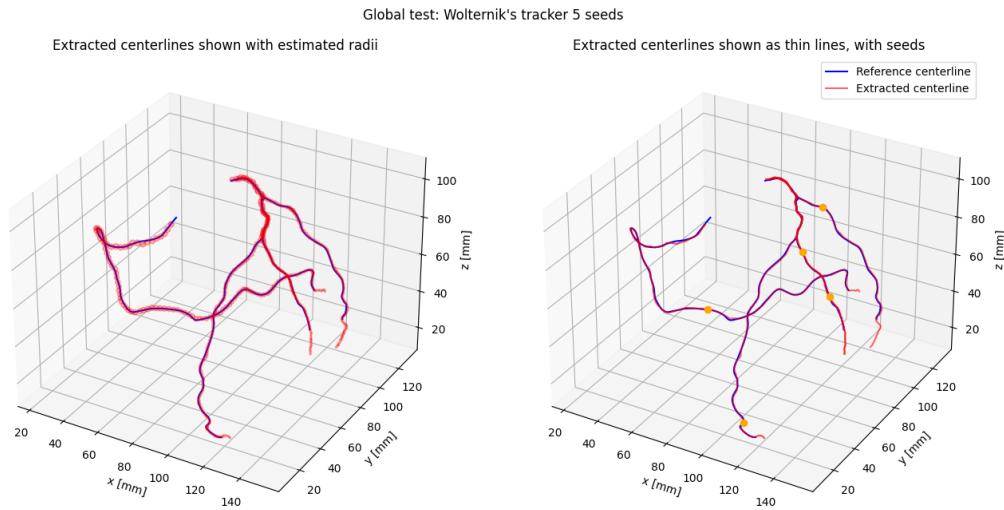


Figure 4.9: Centerlines extracted from 5 seeds with Wolternik's tracker.

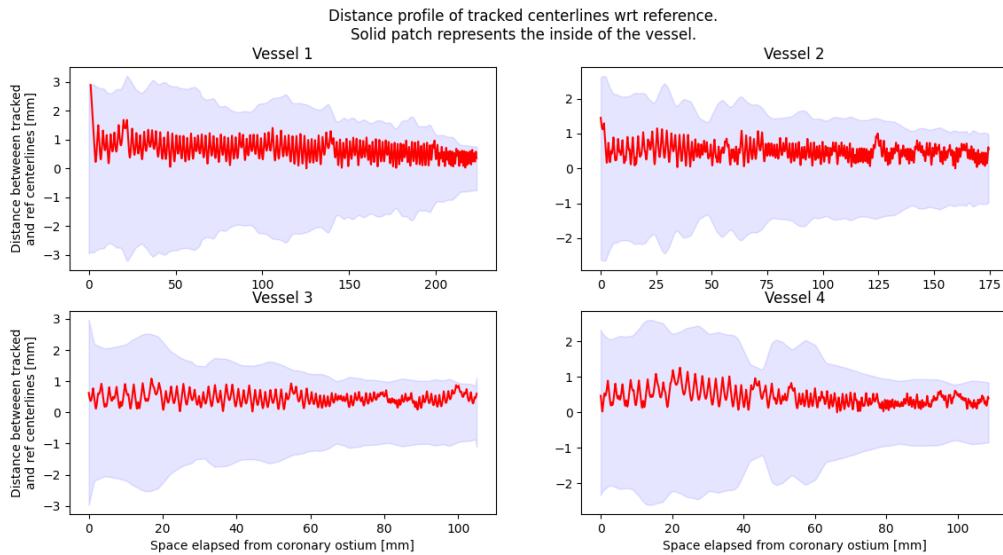


Figure 4.10: Centerlines tracking referred to the reference vessels' centerlines and radii, 5 seeds with Wolternik's tracker [99]. The vessel's reference centerline is straightened on the $y = 0$ axis of the figure, while the shadowed patch represents the inside of the vessel [72]. Tracking seems wiggly just because of the method used to compute connections between reference and extracted centerlines.

The second tracker to be tested is Wolternik's tracker [99]. The tracker performed overall quite well in the 5 seeds challenge (figure 4.9 and 4.10, table 4.4), however it performed insufficiently in the 2 seeds challenge (figure 4.11 and 4.12). The two arteries branching from the left main artery are not explored. 4.8).

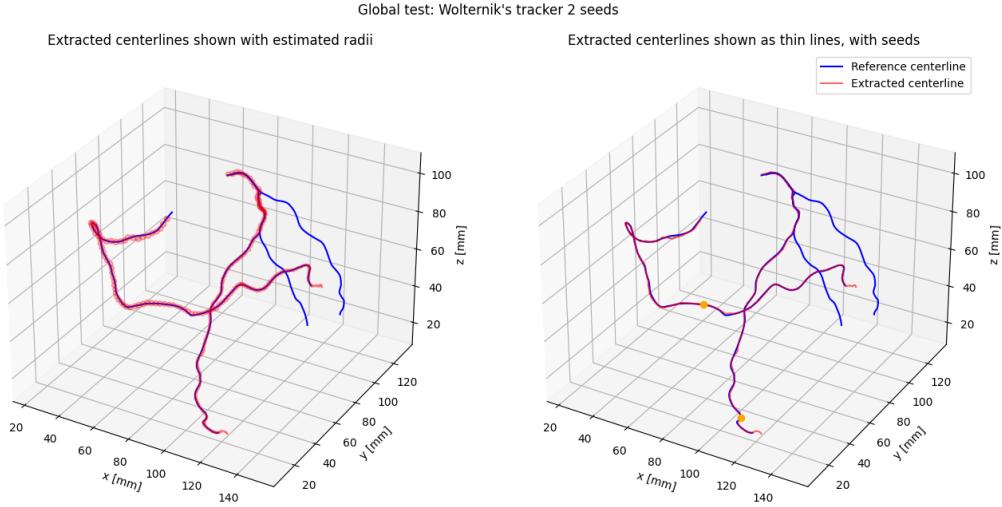


Figure 4.11: Centerlines extracted from 2 seeds with Wolternik's tracker [99]. Two of the four reference vessels are completely ignored.

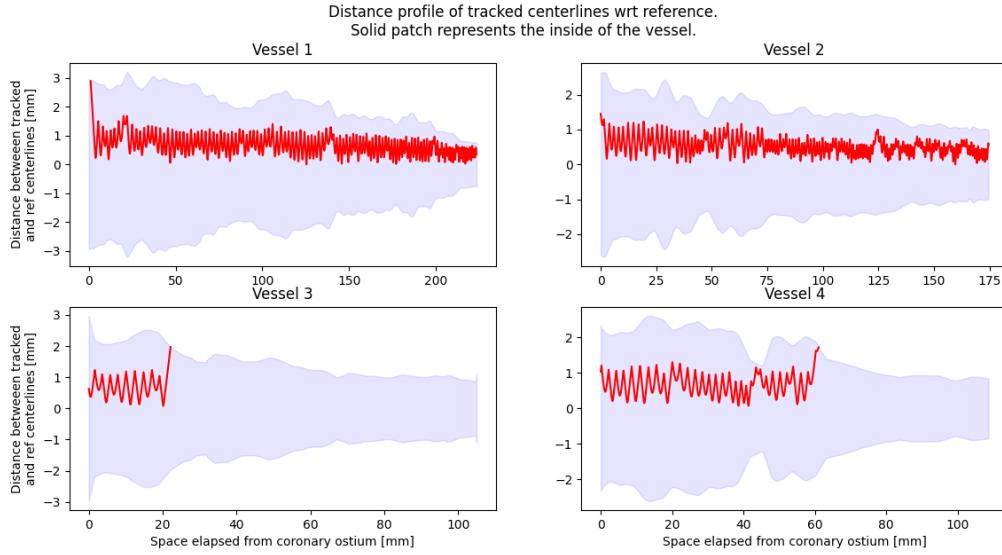


Figure 4.12: Centerlines tracking referred to the reference vessels' centerlines and radii, 2 seeds with Wolternik's tracker [99]. From this graphs it is evident that two vessels were not correctly mapped.

The final test is performed on the seedspawner tracker. It tracked all the centerlines with ease starting from 5 seeds, but it is with 2 initial seeds that it shows its full potential. In figure 4.13 and 4.14 are reported the results of the tracking starting from 2 seeds. The proposed tracker is able to track all the reference centerlines fully starting from one seed per each left and right arterial trees, and it does so at a minimal cost in terms of added complexity of the algorithm.

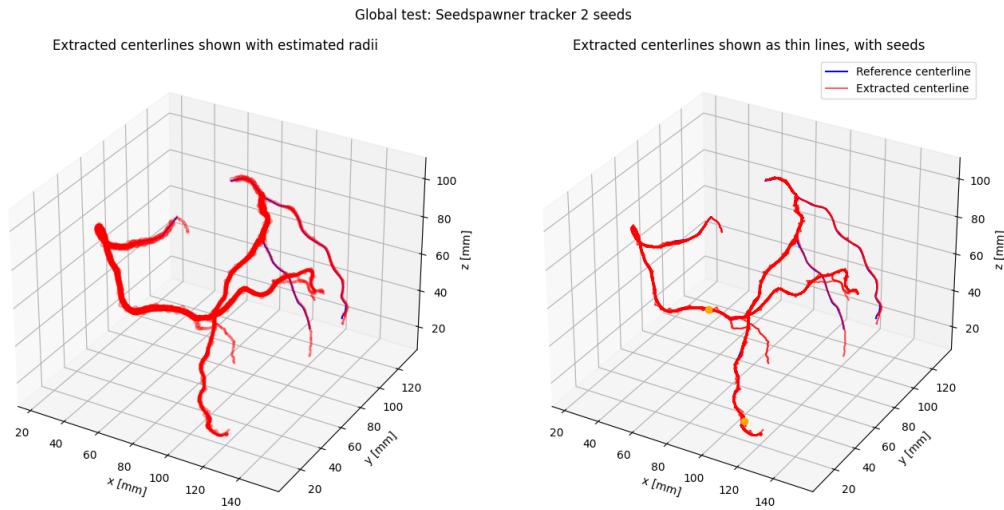


Figure 4.13: Centerlines extracted from 2 seeds with the seedspawner tracker. All reference vessels are explored and tracked, and new uncharted branchings are found in the right arterial tree.

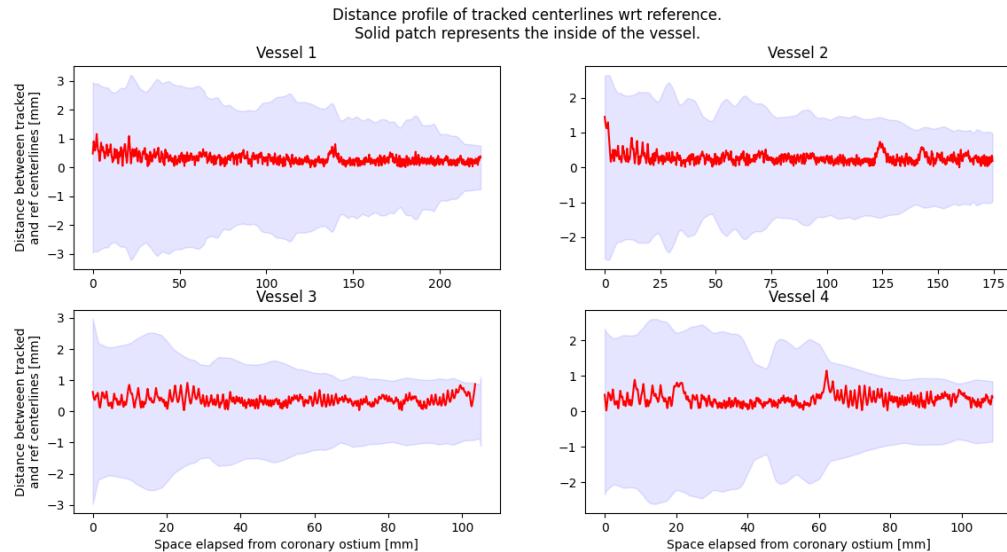


Figure 4.14: Centerlines tracking referred to the reference vessels' centerlines and radii, 2 seeds with the seedspawner tracker. This graph shows that the centerlines are correctly tracked.

In figure 4.15 the planted seed are shown with respect with the original seeds. Planting seeds increases the total tracking time since many seeds have to be considered (from 2 to 200), however it allows to explore the vicinity of the centerlines and discover side branches.

Global test: Seedspawner tracker spawned seeds

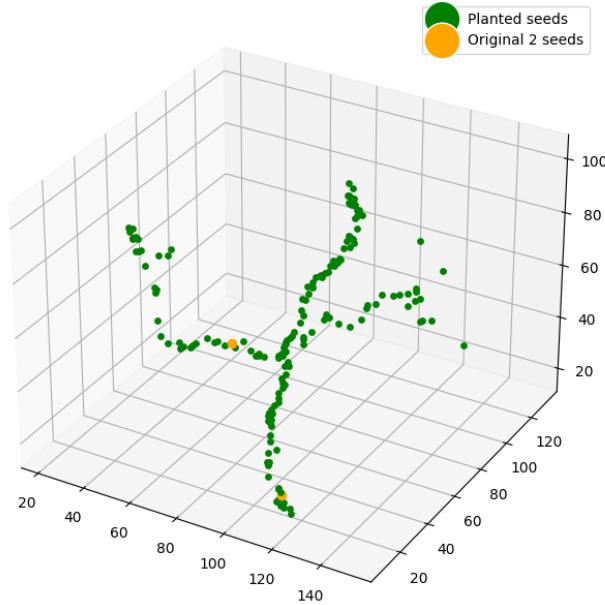


Figure 4.15: Seeds planted by the seedspawner tracker (green) starting from just 2 seeds (yellow).

Table 4.4 proposes a semi-quantitative overview of the tests performed on the trackers.¹

	Basic, 5				Wolternik's, 5				Wolternik's, 2				Seedspawner, 2			
	v1	v2	v3	v4	v1	v2	v3	v4	v1	v2	v3	v4	v1	v2	v3	v4
OV %	62.56	60.13	97.6	100	99.52	100	100	100	99.52	100	21.11	56.21	100	100	98.6	100
OF %	62.56	37.66	97.6	100	0 *	100	100	100	0 *	100	21.11	56.21	100	100	98.6	100
OF [mm]	140	66	103	109	0 *	176	106	109	0 *	176	22	61	224	176	104	109
OT %	71	69.8	100	100	99.45	100	100	100	99.45	100	42	97.8	100	100	100	100
AI [mm]	0.846	0.686	0.603	0.568	0.657	0.474	0.461	0.433	0.657	0.51	0.698	0.670	0.301	0.267	0.365	0.342

Table 4.4: Table reporting the computed metrics for each test, for each vessel. The blue metrics give an idea of the tracking ability, while the orange metric AI of the tracking accuracy.

¹Values of 0 in OF are encountered when, right from the start (coronary ostium) the tracked centerline is out of the reference centerline radius. It means that the tracker deviates from the correct course when it gets near the coronary ostium. This phenomenon was not observed in the Seedspawner tracker.

5 | Conclusions and future developments

An automatic deep learning based coronary artery centerline tracker in CCTA images was proposed, as well as a new CNN architectures to drive the proposed tracker which surpasses the performance of the CNN from Wolternik et al. [99]. The framework was evaluated locally through two disjointed test suites, one for testing just the CNN performance, the other for testing and comparing different trackers similarly to how tests are performed in the Rotterdam Standard Coronary Artery Extraction Evaluation Framework [72].

Three new neural networks were proposed starting from the assumption that the network proposed by Wolternik et al. [99] is undersized and that the information flow through the subsequent layers of the deep neural networks could be enhanced. The first proposed network (input-layer-split) was just composed of two identical network's architectures like the one proposed by Wolternik [99], where each network branch was trained specifically to perform either direction classification or radius estimation through regression. Tests showed that this modification led to an increase in performances, however the network was double the size of the baseline network from Wolternik [99]. The second proposed network (midsplit-4) keeps the first 4 layers (2 standard convolutional and 2 dilated) in common between the two branches, and then assigned one specific task per each branch. The inspiration leading this design choice is found in the structure of the visual cortex of mammals and in many network's architecture found in recent literature, where the first few layers are used to extract low-level information such as edges or corners and curves, while the last layers are used to derive problem specific knowledge from information passed from the previous layers. In this network, information about the estimated direction has no way to flow to the radius estimation branch, which was deemed as a possible limiting factor. A third network was proposed (midsplit-4-FD, Feedback on Directions) where the direction classification output is passed to the middle of the radius estimation branch (after layer 5 of 8). Also, another fully connected layer is added to the end of the radius estimation branch to better account for possible non-linearities introduced by the direction

classification input. The moving force towards this direction was to enhance the radius estimation capability of the network since it was found to be a major error contributor to the closeness of the final predicted landing point and the annotated centerline. The proposed best-performing network midsplit-4-FD can place the landing point 14% closer on average to the reference centerline with respect to Wolternik’s network [99], and it also learns to get 17.9% closer on average to the optimal landing point the network was trained on. It also shows a more consistent behaviour, as the standard deviations of the computed metrics decreased as well.

Furthermore, three trackers were created and discussed. A basic tracker was created in python through object-oriented programming paradigm. This allows the basic tracker’s functionalities to be extended very quickly, enhancing the effectiveness and speed of the research workflow. Any module of the tracker can be extended and modified, and any neural network or any other tracking model can be plugged in instantly. The tracker proposed by Wolternik [99] is re-implemented on the shoulders of the basic tracker. This tracker needs a high number of seedspoint to extract the full coronary artery tree at least in the annotated branches, with a minimum of 1 seed per each vessel sub-branch (4 seeds per image). The proposed seedspawner tracker extends the functionality of Wolternik’s tracker by implementing two different methods to plant seeds along the way while tracking, in an Hansel and Gretel fashion. Global tests performed on the three trackers showed in a qualitative and rough-quantitative way that the proposed tracker, however simple, is capable of extracting the entire (left or right) clinically relevant coronary tree from a single seed planted in the left and right arterial branches in a CCTA image from the CAT08 challenge, while Wolternik’s tracker [99] fails at the task. The accuracy measures AI of the global test suite passed from an average of $\bar{AI} = 0.63$ mm for Wolternik’s tracker to $\bar{AI} = 0.32$ for the seedspawner tracker. The introduction of new seeds in the queue planted noisily along the way allows to explore section nearby the centerline and, in case of nearby bifurcation, to enter and explore a new, unexplored vessel of the coronary artery tree.

The local and global tests performed in a standardised way for both the baseline CNN and tracker proposed by Wolternik [99] re-implemented locally, and for the proposed CNN (midsplit-4-FD) and tracker (seedspawner) showed that both the proposed CNN and the proposed tracker score higher performance indexes with respect to the re-implemented baseline.

In future work, the proposed CNN and tracker will be tested inside the Rotterdam Stan-

dard Evaluation Framework, allowing comparison with other recently developed tracker approaches.

Future developments of the proposed work can involve every aspect of the discussed pipeline: the CNN, the training data and training strategy, the tracker, and the post-processing step.

One promising development for the future of the CNN is to implement a recurrent neural networks as the core network of the tracker to retain a state for the expected intensity values along the current centerline. Thus would be beneficial for several reasons: retaining intensity values along the current centerline allows the tracker to not overstep into other nearby vessels, and retaining the previous state would increase radius and direction predictions consistency, resulting in a smoother and better tracker. Recurrent CNNs can be also employed to perform other tasks inside the tracker, such as movement termination logic based not just on the current step, but with short-term memory, and bifurcation detection to help the tracker explore more of the arterial tree. Thanks to the highly modular structure of the developed tracker architecture, any new module can be quite simply plugged in and interfaced with the rest of the tracker, allowing for faster prototyping with respect with strictly procedural algorithms.

Training data have to be increased in number and variety with respect with just 8 training images available in the CAT08 dataset. All recent state-of-the-art approaches based on deep learning tracking algorithms can count on a high number of privately annotated images, from 50 up to 100, which would drastically increase the training effectiveness.

New training strategies can be inspired by recent developments in reinforcement learning. The full tracker can be trained for many generations by starting on a random point on an annotated centerline and has to learn how best to get to an endpoint by minimising the distance from the annotated centerline. the core network can be pre-trained, and refinements can be performed in this framework. In this way, the network would intrinsically learn the best course of actions also depending on the tracker's characteristics and logic. The proposed post-processing approach is yet incomplete and the endpoints detection task is still quite frail in discerning true endpoints from noisy in-centerline points. Also the research of the shortest patch between the coronary ostium and the coronary branch endpoints can be enhanced by considering a cost function that includes information on the spatial distribution of the processed nodes, to avoid cutting the path and getting furthest away from the reference centerline instead of nearer. A force field-based approach was already tested, with no success.

Overall, the proposed tracking scheme improves the one proposed by Wolternik et al. in 2018 [99] by proposing a better performing network tested locally and implement a

very simple extension to the tracker basic tracker logic for discovering new viable paths by planting seeds along the way while tracking the vessel’s centerline, as opposed to the more elegant yet complicated bifurcation classification network proposed by Salahuddin et al. in 2021 [69], which needs a new set of annotated data to learn to detect bifurcations.

In the near future, hopefully the proposed trackers will be the basis for the coronary artery extraction system used in the fully automatic estimation of the Fractional Flow Reserve index non-invasively from CCTA scans. Hopefully, with enough training data, the new FFR_{CT} estimation approaches can be trained on CCTA which do not even require the hyperaemia condition.

Bibliography

- [1] Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin. *Learning from data*, volume 4. AMLBook New York, NY, USA:, 2012.
- [2] S. Achenbach, T. Rudolph, J. Rieber, H. Eggebrecht, G. Richardt, T. Schmitz, N. Werner, F. Boenner, and H. Möllmann. Performing and interpreting fractional flow reserve measurements in clinical practice: An expert consensus document. *Interventional Cardiology Review*, 12:97, 09 2017. doi: 10.15420/icr.2017:13:2. PMID: 29588737, PMCID: PMC5808579.
- [3] Y. Arad, L. Spadaro, K. Goodman, D. Newstein, and A. Guerci. Prediction of coronary events with electron beam computed tomography. *Journal of the American College of Cardiology*, 36:1253–60, 11 2000. doi: 10.1016/S0735-1097(00)00872-X.
- [4] L. Athanasiou, C. Bourantas, P. Siogkas, A. Sakellarios, T. Exarchos, K. NAKA, M. Papafakis, L. Michalis, F. Prati, and D. Fotiadis. 3d reconstruction of coronary arteries using frequency domain optical coherence tomography images and biplane angiography. *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 2012:2647–50, 08 2012. doi: 10.1109/EMBC.2012.6346508.
- [5] L. Athanasiou, G. Rigas, A. Sakellarios, T. Exarchos, P. Siogkas, C. Bourantas, H. Garcia-Garcia, P. Lemos, B. Falcão, L. Michalis, O. Parodi, f. vozzi, and D. Fotiadis. Three-dimensional reconstruction of coronary arteries and plaque morphology using ct angiography – comparison and registration with ivus. *BMC Medical Imaging*, 16, 01 2016. doi: 10.1186/s12880-016-0111-6.
- [6] S. Aylward and E. Bullitt. Initialization, noise, singularities, and scale in height ridge traversal for tubular object centerline extraction. *IEEE Transactions on Medical Imaging*, 21(2):61–75, 2002. doi: 10.1109/42.993126.
- [7] E. V. Belle, G. Rioufol, C. Pouillot, T. Cuisset, K. Bougrini, E. Teiger, S. Champagne, L. Belle, D. Barreau, M. Hanssen, C. Besnard, R. Dauphin, J. Dallongeville,

- Y. E. Hahi, G. Sideris, C. Bretelle, N. Lhoest, P. Barnay, L. Leborgne, and P. Dupouy. Outcome impact of coronary revascularization strategy reclassification with fractional flow reserve at time of diagnostic angiography. *Circulation*, 129(2):173–185, 2014. doi: 10.1161/CIRCULATIONAHA.113.006646. URL <https://www.ahajournals.org/doi/abs/10.1161/CIRCULATIONAHA.113.006646>.
- [8] E. J. Benjamin, S. S. Virani, C. W. Callaway, A. M. Chamberlain, A. R. Chang, S. Cheng, S. E. Chiuve, M. Cushman, F. N. Delling, R. Deo, S. D. de Ferranti, J. F. Ferguson, M. Fornage, C. Gillespie, C. R. Isasi, M. C. Jiménez, L. C. Jordan, S. E. Judd, D. Lackland, J. H. Lichtman, L. Lisabeth, S. Liu, C. T. Longenecker, P. L. Lutsey, J. S. Mackey, D. B. Matchar, K. Matsushita, M. E. Mussolino, K. Nasir, M. O’Flaherty, L. P. Palaniappan, A. Pandey, D. K. Pandey, M. J. Reeves, M. D. Ritchey, C. J. Rodriguez, G. A. Roth, W. D. Rosamond, U. K. Sampson, G. M. Satou, S. H. Shah, N. L. Spartano, D. L. Tirschwell, C. W. Tsao, J. H. Voeks, J. Z. Willey, J. T. Wilkins, J. H. Wu, H. M. Alger, S. S. Wong, and P. Muntner. Heart disease and stroke statistics—2018 update: A report from the american heart association. *Circulation (New York, N.Y.)*, 137(12):e67–e492, 2018. ISSN 0009-7322.
- [9] C. Bourantas, F. Kalatzis, M. Papafaklis, D. Fotiadis, A. Tweddel, I. Kourtis, C. Katsouras, and L. Michalis. Angiocare: An automated system for fast three-dimensional coronary reconstruction by integrating angiographic and intracoronary ultrasound data. *Catheterization and Cardiovascular Interventions*, 72:166 – 175, 08 2008. doi: 10.1002/ccd.21527.
- [10] C. V. Bourantas, I. C. Kourtis, M. E. Plissiti, D. I. Fotiadis, C. S. Katsouras, M. I. Papafaklis, and L. K. Michalis. A method for 3d reconstruction of coronary arteries using biplane angiography and intravascular ultrasound images. *Computerized Medical Imaging and Graphics*, 29(8):597–606, 2005. ISSN 0895-6111. doi: <https://doi.org/10.1016/j.compmedimag.2005.07.001>. URL <https://www.sciencedirect.com/science/article/pii/S089561105000777>.
- [11] B. Bouraoui, C. Ronse, J. Baruthio, N. Passat, and P. Germain. Fully automatic 3d segmentation of coronary arteries based on mathematical morphology. In *2008 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 1059–1062, 2008. doi: 10.1109/ISBI.2008.4541182.
- [12] S. Cetin and G. Unal. A higher-order tensor vessel tractography for segmentation of vascular structures. *IEEE Transactions on Medical Imaging*, 34(10):2172–2185, 2015. doi: 10.1109/TMI.2015.2425535.

- [13] C. Chen, C. Qin, H. Qiu, G. Tarroni, J. Duan, W. Bai, and D. Rueckert. Deep learning for cardiac image segmentation: A review. *Frontiers in Cardiovascular Medicine*, 7:25, 2020. ISSN 2297-055X. doi: 10.3389/fcvm.2020.00025. URL <https://www.frontiersin.org/article/10.3389/fcvm.2020.00025>.
- [14] Y.-C. Chen, Y.-C. Lin, C.-P. Wang, C.-Y. Lee, W.-J. Lee, T.-D. Wang, and C.-M. Chen. Coronary artery segmentation in cardiac ct angiography using 3d multi-channel u-net. *arXiv preprint arXiv:1907.12246*, 2019.
- [15] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger. 3d u-net: Learning dense volumetric segmentation from sparse annotation. In S. Ourselin, L. Joskowicz, M. R. Sabuncu, G. Unal, and W. Wells, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, pages 424–432, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46723-8.
- [16] A. Coenen, Y.-H. Kim, M. Kruk, C. Tesche, J. De Geer, A. Kurata, M. Lubbers, J. Daemen, L. M. Itu, S. Rapaka, P. Sharma, C. Schwemmer, A. Persson, U. Schoepf, C. Kepka, D. Yang, and K. Nieman. Diagnostic accuracy of a machine-learning approach to coronary computed tomographic angiography-based fractional flow reserve: Result from the machine consortium. *Circulation: Cardiovascular Imaging*, 11:e007217, 06 2018. doi: 10.1161/CIRCIMAGING.117.007217.
- [17] R. Damalerio, E. L. Tan, K. L. Tan, R. Lim, W. Chen, and M.-y. Cheng. Ultra slim packaging and assembly method for 360- μm diameter guide wires. In *2015 IEEE 17th Electronics Packaging and Technology Conference (EPTC)*, pages 1–4, 2015. doi: 10.1109/EPTC.2015.7727915.
- [18] R. C. Deo. Machine learning in medicine. *Circulation*, 132(20):1920–1930, 2015. doi: 10.1161/CIRCULATIONAHA.115.001593. URL <https://www.ahajournals.org/doi/abs/10.1161/CIRCULATIONAHA.115.001593>.
- [19] P. Douglas, G. Pontone, M. Hlatky, M. Patel, B. Nørgaard, R. Byrne, N. Curzen, I. Purcell, M. Gutberlet, G. Rioufol, U. Hink, H. Schuchlenz, G. Feuchtner, M. Gilard, D. Andreini, J. Jensen, M. Hadamitzky, K. Chiswell, D. Cyr, and B. Bruyne. Clinical outcomes of fractional flow reserve by computed tomographic angiography-guided diagnostic strategies vs. usual care in patients with suspected coronary artery disease: The prospective longitudinal trial of ffrct: Outcome and resource impacts study. *European Heart Journal*, 36:ehv444, 09 2015. doi: 10.1093/eurheartj/ehv444.
- [20] Y. Duan, J. Feng, J. Lu, and J. Zhou. Context aware 3d fully convolutional networks

- for coronary artery segmentation. In *International Workshop on Statistical Atlases and Computational Models of the Heart*, pages 85–93. Springer, 2018.
- [21] O. Ecabert, J. Peters, H. Schramm, C. Lorenz, J. von Berg, M. J. Walker, M. Vembar, M. E. Olszewski, K. Subramanyan, G. Lavi, and J. Weese. Automatic model-based segmentation of the heart in CT images. *IEEE Trans Med Imaging*, 27(9):1189–1201, Sept. 2008.
 - [22] I. Encyclopædia Britannica. Heart. <https://www.britannica.com/science/heart>, June 2021. The Editors of Encyclopædia Britannica.
 - [23] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. pages 226–231. AAAI Press, 1996.
 - [24] W. F. Fearon, B. Bornschein, P. A. Tonino, R. M. Gothe, B. D. Bruyne, N. H. Pijls, and U. Siebert. Economic evaluation of fractional flow reserve-guided percutaneous coronary intervention in patients with multivessel disease. *Circulation*, 122(24):2545–2550, 2010.
 - [25] O. Friman, M. Hindennach, C. Kühnel, and H.-O. Peitgen. Multiple hypothesis template tracking of small 3d vessel structures. *Medical Image Analysis*, 14(2):160–171, 2010. ISSN 1361-8415. doi: <https://doi.org/10.1016/j.media.2009.12.003>. URL <https://www.sciencedirect.com/science/article/pii/S1361841509001479>.
 - [26] R. Gao, Z. Hou, J. Li, H. Han, B. Lu, and S. K. Zhou. Joint coronary centerline extraction and lumen segmentation from ccta using cnntracker and vascular graph convolutional network. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 1897–1901, 2021. doi: 10.1109/ISBI48211.2021.9433764.
 - [27] S. Gauss, S. Achenbach, T. Pflederer, A. Schuhbaeck, W. Daniel, and M. Mohamed. Assessment of coronary artery remodelling by dual-source ct: A head-to-head comparison with intravascular ultrasound. *Heart (British Cardiac Society)*, 97:991–7, 06 2011. doi: 10.1136/hrt.2011.223024.
 - [28] T. Gori and M. Fineschi, editors. *Atlas of FFR-Guided Percutaneous Coronary Interventions*. Springer International Publishing, 2016. doi: 10.1007/978-3-319-47116-7. URL <https://doi.org/10.1007%2F978-3-319-47116-7>.
 - [29] M. A. Gülsün, G. Funka-Lea, P. Sharma, S. Rapaka, and Y. Zheng. Coronary centerline extraction via optimal flow paths and cnn path pruning. In S. Ourselin, L. Joskowicz, M. R. Sabuncu, G. Unal, and W. Wells, editors, *Medical Image Com-*

- puting and Computer-Assisted Intervention - MICCAI 2016*, pages 317–325, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46726-9.
- [30] P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968. doi: 10.1109/TSSC.1968.300136.
 - [31] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.
 - [32] W. Huang, L. Huang, Z. Lin, S. Huang, Y. Chi, J. Zhou, J. Zhang, R.-S. Tan, and L. Zhong. Coronary artery segmentation by deep learning neural networks on computed tomographic coronary angiographic images. In *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 608–611, 2018. doi: 10.1109/EMBC.2018.8512328.
 - [33] W. Huang, L. Huang, Z. Lin, S. Huang, Y. Chi, J. Zhou, J. Zhang, R.-S. Tan, and L. Zhong. Coronary artery segmentation by deep learning neural networks on computed tomographic coronary angiographic images. In *2018 40th Annual international conference of the IEEE engineering in medicine and biology society (EMBC)*, pages 608–611. IEEE, 2018.
 - [34] F. Isensee, P. F. Jaeger, S. A. A. Kohl, J. Petersen, and K. H. Maier-Hein. nnunet: a self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods*, 18(2):203–211, Feb 2021. ISSN 1548-7105. doi: 10.1038/s41592-020-01008-z. URL <https://doi.org/10.1038/s41592-020-01008-z>.
 - [35] L. Itu, P. Sharma, V. Mihalef, A. Kamen, C. Suciu, and D. Comaniciu. A patient-specific reduced-order model for coronary circulation. In *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*, pages 832–835, 2012. doi: 10.1109/ISBI.2012.6235677.
 - [36] L. Itu, S. Rapaka, T. Passerini, B. Georgescu, C. Schwemmer, M. Schoeberger, T. Flohr, P. Sharma, and D. Comaniciu. A machine-learning approach for computation of fractional flow reserve from coronary computed tomography. *Journal of Applied Physiology*, 121(1):42–52, 2016. doi: 10.1152/japplphysiol.00752.2015. URL <https://journals.physiology.org/doi/abs/10.1152/japplphysiol.00752.2015>. PMID: 27079692.
 - [37] B. Jeon, Y. Jang, H. Shim, and H.-J. Chang. Identification of coronary arteries in ct images by bayesian analysis of geometric relations among anatomical landmarks. *Pattern Recognition*, 96:106958, 2019. ISSN 0031-3203. doi:

- <https://doi.org/10.1016/j.patcog.2019.07.003>. URL <https://www.sciencedirect.com/science/article/pii/S0031320319302559>.
- [38] D. Kang, J. Woo, C. J. Kuo, P. J. Slomka, D. Dey, and G. Germano. Heart chambers and whole heart segmentation techniques. *Journal of Electronic Imaging*, 21(1):010901, 2012.
 - [39] A. Kerkeni, A. Benabdallah, A. Manzanera, and M. H. Bedoui. A coronary artery segmentation method based on multiscale analysis and region growing. *Computerized Medical Imaging and Graphics*, 48:49–61, 2016. ISSN 0895-6111. doi: <https://doi.org/10.1016/j.compmedimag.2015.12.004>. URL <https://www.sciencedirect.com/science/article/pii/S0895611115001913>.
 - [40] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
 - [41] C. Kirbas and F. Quek. A review of vessel extraction techniques and algorithms. *ACM Computing Surveys*, 36, 12 2002. doi: 10.1145/1031120.1031121.
 - [42] H. Kirişli, M. Schaap, C. Metz, A. Dharampal, W. Meijboom, S. Papadopoulou, A. Dedic, K. Nieman, M. de Graaf, M. Meijs, M. Cramer, A. Broersen, S. Cetin, A. Eslami, L. Flórez-Valencia, K. Lor, B. Matuszewski, I. Melki, B. Mohr, I. Öksüz, R. Shahzad, C. Wang, P. Kitslaar, G. Unal, A. Katouzian, M. Orkisz, C. Chen, F. Precioso, L. Najman, S. Masood, D. Ünay, L. van Vliet, R. Moreno, R. Goldenberg, E. Vuçini, G. Krestin, W. Niessen, and T. van Walsum. Standardized evaluation framework for evaluating coronary artery stenosis detection, stenosis quantification and lumen segmentation algorithms in computed tomography angiography. *Medical Image Analysis*, 17(8):859–876, 2013. ISSN 1361-8415. doi: <https://doi.org/10.1016/j.media.2013.05.007>. URL <https://www.sciencedirect.com/science/article/pii/S1361841513000819>.
 - [43] B. Ko, J. Cameron, R. Munnur, D. Wong, Y. Fujisawa, T. Sakaguchi, K. Hirohata, J. Hislop-Jambrich, S. Fujimoto, K. Takamura, M. Crossett, M. Leung, A. Kuganesan, Y. Malaiapan, A. Nasis, J. Troupis, P. I. Meredith AM, and S. Seneviratne. Noninvasive ct-derived ffr based on structural and fluid analysis. *JACC: Cardiovascular Imaging*, 10, 10 2016. doi: 10.1016/j.jcmg.2016.07.005.
 - [44] Y. Kobayashi, N. P. Johnson, F. M. Zimmermann, N. Witt, C. Berry, A. Jeremias, B.-K. Koo, G. Esposito, G. Rioufol, S.-J. Park, T. Nishi, D.-H. Choi, K. G. Oldroyd, E. Barbato, N. H. Pijls, B. De Bruyne, and W. F. Fearon. Agreement of the resting distal to aortic coronary pressure with the instantaneous wave-

- free ratio. *Journal of the American College of Cardiology*, 70(17):2105–2113, 2017. ISSN 0735-1097. doi: <https://doi.org/10.1016/j.jacc.2017.08.049>. URL <https://www.sciencedirect.com/science/article/pii/S0735109717393683>.
- [45] K. Krissian, H. Bogunovic, J. Pozo, M. Villa-Uriol, and A. Frangi. Minimally interactive knowledge-based coronary tracking in cta using a minimal cost path. *The Insight Journal*, 1, 2008.
- [46] M. C. H. Lee, K. Petersen, N. Pawlowski, B. Glocker, and M. Schaap. Tetris: Template transformer networks for image segmentation with shape priors. *IEEE transactions on medical imaging*, 38(11):2596–2606, 2019.
- [47] J. Leipsic, S. Abbara, S. Achenbach, R. Cury, J. P. Earls, G. J. Mancini, K. Nieman, G. Pontone, and G. L. Raff. SCCT guidelines for the interpretation and reporting of coronary CT angiography: a report of the society of cardiovascular computed tomography guidelines committee. *J Cardiovasc Comput Tomogr*, 8(5):342–358, July 2014.
- [48] D. Lesage, E. D. Angelini, I. Bloch, and G. Funka-Lea. A review of 3d vessel lumen segmentation techniques: Models, features and extraction schemes. *Medical Image Analysis*, 13(6):819–845, 2009. ISSN 1361-8415. doi: <https://doi.org/10.1016/j.media.2009.07.011>. URL <https://www.sciencedirect.com/science/article/pii/S136184150900067X>. Includes Special Section on Computational Biomechanics for Medicine.
- [49] S. Leschka, H. Alkadhi, A. Plass, L. Desbiolles, J. Grünenfelder, B. Marincek, and S. Wildermuth. Accuracy of MSCT coronary angiography with 64-slice technology: first experience. *Eur Heart J*, 26(15):1482–1487, Apr. 2005.
- [50] Y. Lian, A. Almasi, D. B. Grayden, T. Kameneva, A. N. Burkitt, and H. Mefbin. Learning receptive field properties of complex cells in v1. *PLoS computational biology*, 17(3):e1007957, 2021.
- [51] M. T. Lu, M. Ferencik, R. S. Roberts, K. L. Lee, A. Ivanov, E. Adami, D. B. Mark, F. A. Jaffer, J. A. Leipsic, P. S. Douglas, and U. Hoffmann. Noninvasive ffr derived from coronary ct angiography: Management and outcomes in the promise trial. *JACC: Cardiovascular Imaging*, 10(11):1350–1358, 2017. ISSN 1936-878X. doi: <https://doi.org/10.1016/j.jcmg.2016.11.024>. URL <https://www.sciencedirect.com/science/article/pii/S1936878X17302620>.
- [52] J. Menke, C. Unterberg-Buchwald, W. Staab, J. M. Sohns, A. Seif Amir Hosseini, and A. Schwarz. Head-to-head comparison of prospectively triggered vs

- retrospectively gated coronary computed tomography angiography: Meta-analysis of diagnostic accuracy, image quality, and radiation dose. *American Heart Journal*, 165(2):154–163.e3, 2013. ISSN 0002-8703. doi: <https://doi.org/10.1016/j.ahj.2012.10.026>. URL <https://www.sciencedirect.com/science/article/pii/S0002870312007363>.
- [53] J. Merkow, A. Marsden, D. Kriegman, and Z. Tu. Dense volume-to-volume vascular boundary detection. In *International conference on medical image computing and computer-assisted intervention*, pages 371–379. Springer, 2016.
 - [54] D. N. Metaxas, L. Axel, G. Fichtinger, and G. Székely, editors. *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2008, 11th International Conference, New York, NY, USA, September 6-10, 2008, Proceedings, Part I*, volume 5241 of *Lecture Notes in Computer Science*, 2008. Springer. ISBN 978-3-540-85987-1. doi: 10.1007/978-3-540-85988-8. URL <https://doi.org/10.1007/978-3-540-85988-8>.
 - [55] D. N. Metaxas, L. Axel, G. Fichtinger, and G. Székely, editors. *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2008, 11th International Conference, New York, NY, USA, September 6-10, 2008, Proceedings, Part II*, volume 5242 of *Lecture Notes in Computer Science*, 2008. Springer. ISBN 978-3-540-85989-5. doi: 10.1007/978-3-540-85990-1. URL <https://doi.org/10.1007/978-3-540-85990-1>.
 - [56] G. S. Mintz, S. E. Nissen, W. D. Anderson, S. R. Bailey, R. Erbel, P. J. Fitzgerald, F. J. Pinto, K. Rosenfield, R. J. Siegel, E. Tuzcu, P. G. Yock, R. A. O’Rourke, J. Abrams, E. R. Bates, B. R. Brodie, P. S. Douglas, G. Gregoratos, M. A. Hlatky, J. S. Hochman, S. Kaul, C. M. Tracy, D. D. Waters, and W. L. Winters. American college of cardiology clinical expert consensus document on standards for acquisition, measurement and reporting of intravascular ultrasound studies (ivus)³. *Journal of the American College of Cardiology*, 37(5):1478–1492, 2001. doi: 10.1016/S0735-1097(01)01175-5. URL <https://www.jacc.org/doi/abs/10.1016/S0735-1097%2801%2901175-5>.
 - [57] P. Moeskops, J. M. Wolterink, B. H. van der Velden, K. G. Gilhuijs, T. Leiner, M. A. Viergever, and I. Išgum. Deep learning for multi-task medical image segmentation in multiple modalities. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 478–486. Springer, 2016.
 - [58] S. Moore, K. Halupka, and S. Zhuk. Using observed contrast motion during cardiac

- catheterization to tune patient specific coronary blood flow simulations. In *2018 Computing in Cardiology Conference (CinC)*, volume 45, pages 1–4, 2018. doi: 10.22489/CinC.2018.235.
- [59] M. Motwani, D. Dey, D. Berman, G. Germano, S. Achenbach, M. Al-Mallah, D. Andreini, M. Budoff, F. Cademartiri, T. Callister, H.-J. Chang, K. Chinnaian, B. Chow, R. Cury, A. Delago, M. Gomez, H. Gransar, M. Hadamitzky, J. Hausleiter, and P. Slomka. Machine learning for prediction of all-cause mortality in patients with suspected coronary artery disease: A 5-year multicentre prospective registry analysis. *European Heart Journal*, 38:ehw188, 06 2016. doi: 10.1093/eurheartj/ehw188.
- [60] H. Nickisch, Y. Lamash, S. Prevrhal, M. Freiman, M. Vembar, L. Goshen, and H. Schmitt. Learning patient-specific lumped models for interactive coronary blood flow simulations. In N. Navab, J. Hornegger, W. M. Wells, and A. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 433–441, Cham, 2015. Springer International Publishing. ISBN 978-3-319-24571-3.
- [61] B. L. Nørgaard, J. Leipsic, S. Gaur, S. Seneviratne, B. S. Ko, H. Ito, J. M. Jensen, L. Mauri, B. De Bruyne, H. Bezerra, K. Osawa, M. Marwan, C. Naber, A. Erglis, S.-J. Park, E. H. Christiansen, A. Kaltoft, J. F. Lassen, H. E. Bøtker, and S. Achenbach. Diagnostic performance of noninvasive fractional flow reserve derived from coronary computed tomography angiography in suspected coronary artery disease: The nxt trial (analysis of coronary blood flow using ct angiography: Next steps). *Journal of the American College of Cardiology*, 63(12):1145–1155, 2014. ISSN 0735-1097. doi: <https://doi.org/10.1016/j.jacc.2013.11.043>. URL <https://www.sciencedirect.com/science/article/pii/S073510971400165X>.
- [62] S.-L. Papadopoulou, L. A. Neefjes, M. Schaap, H.-L. Li, E. Capuano, A. G. van der Giessen, J. C. Schuurbiers, F. J. Gijsen, A. S. Dharampal, K. Nieman, R. J. van Geuns, N. R. Mollet, and P. J. de Feyter. Detection and quantification of coronary atherosclerotic plaque by 64-slice multidetector ct: A systematic head-to-head comparison with intravascular ultrasound. *Atherosclerosis*, 219(1):163–170, 2011. ISSN 0021-9150. doi: <https://doi.org/10.1016/j.atherosclerosis.2011.07.005>. URL <https://www.sciencedirect.com/science/article/pii/S0021915011005843>.
- [63] S.-L. Papadopoulou, C. Girasis, A. Dharampal, V. Farooq, Y. Onuma, A. Rossi, M. angèle Morel, G. P. Krestin, P. W. Serruys, P. J. de Feyter, and H. M. G. Garcia. Ct-syntax score. *JACC: Cardiovascular Imaging*, 6(3):413–415, 2013. doi:

- 10.1016/j.jcmg.2012.09.013. URL <https://www.jacc.org/doi/abs/10.1016/j.jcmg.2012.09.013>.
- [64] L. Papamanolis, H. Kim, C. Jaquet, M. Sinclair, M. Schaap, I. Danad, P. van Diemen, P. Knaapen, L. Najman, H. Talbot, C. Taylor, and I. Vignon-Clementel. Myocardial perfusion simulation for coronary artery disease: A coupled patient-specific multiscale model. *Annals of Biomedical Engineering*, 12 2020. doi: 10.1007/s10439-020-02681-z.
- [65] S.-J. Park, S.-J. Kang, J.-M. Ahn, E. Shim, Y.-T. Kim, S.-C. Yun, H. G. Song, J.-Y. Lee, W.-J. Kim, D.-W. Park, S.-W. Lee, Y.-H. Kim, C. Lee, G. Mintz, and S.-W. Park. Visual-functional mismatch between coronary angiography and fractional flow reserve. *JACC. Cardiovascular interventions*, 5:1029–36, 10 2012. doi: 10.1016/j.jcin.2012.07.007.
- [66] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017.
- [67] N. H. Pijls, P. van Schaardenburgh, G. Manoharan, E. Boersma, J.-W. Bech, M. van't Veer, F. Bär, J. Hoornje, J. Koolen, W. Wijns, et al. Percutaneous coronary intervention of functionally nonsignificant stenosis: 5-year follow-up of the defer study. *Journal of the American College of Cardiology*, 49(21):2105–2111, 2007.
- [68] D. Ropers, J. Rixe, K. Anders, A. Küttner, U. Baum, W. Bautz, W. G. Daniel, and S. Achenbach. Usefulness of multidetector row spiral computed tomography with 64- x 0.6-mm collimation and 330-ms rotation for the noninvasive detection of significant coronary artery stenoses. *Am J Cardiol*, 97(3):343–348, Dec. 2005.
- [69] Z. Salahuddin, M. Lenga, and H. Nickisch. Multi-resolution 3d convolutional neural networks for automatic coronary centerline extraction in cardiac ct angiography scans. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 91–95, 2021. doi: 10.1109/ISBI48211.2021.9434002.
- [70] F. Sant'Anna and L. Sant'Anna. Analysis of ffr measurement clinical impact and cost-effectiveness compared to angiography in multi-arterial patients undergoing pci. *Arquivos Brasileiros de Cardiologia*, 112, 01 2018. doi: 10.5935/abc.20180261.
- [71] F. Sant'Anna, E. Ribeiro, L. Batista, M. Brito, F. Ventura, H. Ferraz, L. Buczynski, C. Barrozo, and N. Pijls. What is the angiography error when defining myocardial ischemia during percutaneous coronary interventions? *Arquivos brasileiros de cardiologia*, 91:162–7, 179, 09 2008.

- [72] M. Schaap, C. T. Metz, T. van Walsum, A. G. van der Giessen, A. C. Weustink, N. R. Mollet, C. Bauer, H. Bogunović, C. Castro, X. Deng, E. Dikici, T. O'Donnell, M. Frenay, O. Friman, M. H. Hoyos, P. H. Kitslaar, K. Krissian, C. Kühnel, M. A. Luengo-Oroz, M. Orkisz, Örjan Smedby, M. Styner, A. Szymczak, H. Tek, C. Wang, S. K. Warfield, S. Zambal, Y. Zhang, G. P. Krestin, and W. J. Niessen. Standardized evaluation methodology and reference database for evaluating coronary artery centerline extraction algorithms. *Medical Image Analysis*, 13(5):701–714, 2009. ISSN 1361-8415. doi: <https://doi.org/10.1016/j.media.2009.06.003>. URL <https://www.sciencedirect.com/science/article/pii/S1361841509000474>. Includes Special Section on the 12th International Conference on Medical Imaging and Computer Assisted Intervention.
- [73] M. Schneider, S. Hirsch, B. Weber, G. Székely, and B. H. Menze. Joint 3-d vessel segmentation and centerline extraction using oblique hough forests with steerable filters. *Medical Image Analysis*, 19(1):220–249, 2015. ISSN 1361-8415. doi: <https://doi.org/10.1016/j.media.2014.09.007>. URL <https://www.sciencedirect.com/science/article/pii/S1361841514001455>.
- [74] J. Schuurbiers, N. Lopez, J. Ligthart, F. Gijsen, J. Dijkstra, P. Serruys, A. van der Steen, and J. Wentzel. In vivo validation of caas qca-3d coronary reconstruction using fusion of angiography and intravascular ultrasound (angus). *Catheterization and cardiovascular interventions : official journal of the Society for Cardiac Angiography & Interventions*, 73:620–6, 04 2009. doi: 10.1002/ccd.21872.
- [75] R. Shahzad, H. Kirişli, C. Metz, H. Tang, M. Schaap, L. Van Vliet, W. Niessen, and T. Walsum. Automatic segmentation, detection and quantification of coronary artery stenoses on cta. *The international journal of cardiovascular imaging*, 29, 08 2013. doi: 10.1007/s10554-013-0271-1.
- [76] R. S. Shantouf and A. Mehra. Coronary fractional flow reserve. *AJR Am J Roentgenol*, 204(3):W261–5, Mar. 2015.
- [77] Y. Shen, Z. Fang, Y. Gao, N. Xiong, C. Zhong, and X. Tang. Coronary arteries segmentation based on 3d fcn with attention gate and level set function. *IEEE Access*, 7:42826–42835, 2019.
- [78] M. Sinclair, A. Schuh, K. Hahn, K. Petersen, Y. Bai, J. Batten, M. Schaap, and B. Glocker. Atlas-istn: Joint segmentation, registration and atlas construction with image-and-spatial transformer networks. *Medical Image Analysis*, 78:102383, 02 2022. doi: 10.1016/j.media.2022.102383.

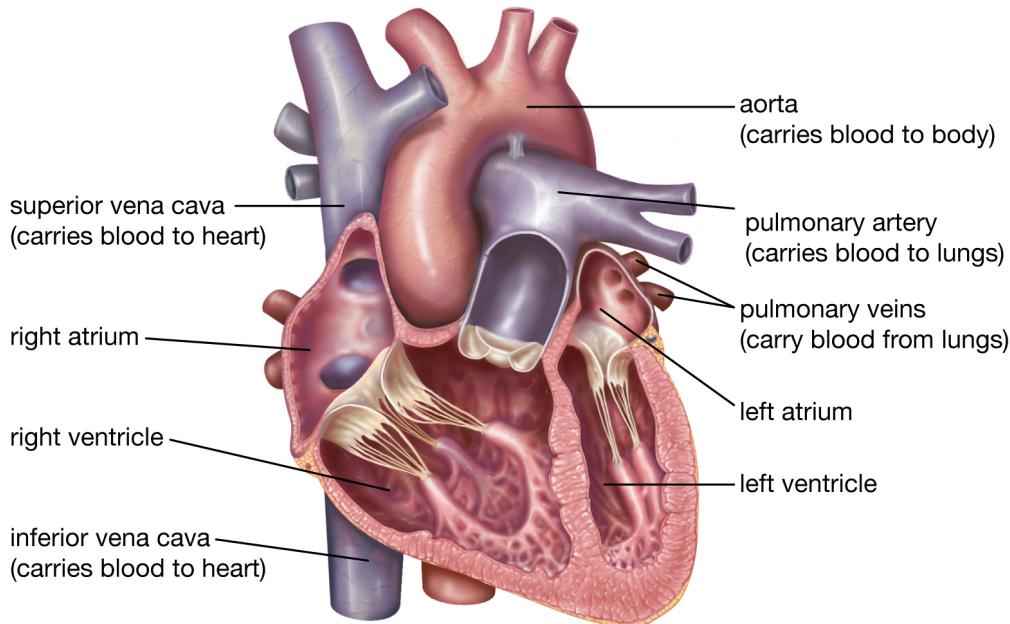
- [79] P. Siogkas, L. Athanasiou, A. Sakellarios, K. Stefanou, T. Exarchos, M. Papafakis, K. NAKA, O. Parodi, L. Michalis, and D. Fotiadis. Validation study of a 3d-qca coronary reconstruction method using a hybrid intravascular ultrasound and angiography reconstruction method and patient-specific fractional flow reserve data. *Conference proceedings: ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 2015:973–6, 08 2015. doi: 10.1109/EMBC.2015.7318526.
- [80] A. Sironi, E. Turetken, V. Lepetit, and P. Fua. Multiscale centerline detection. *IEEE Trans Pattern Anal Mach Intell*, 38(7):1327–1341, July 2016.
- [81] C. A. Taylor and C. Figueira. Patient-specific modeling of cardiovascular mechanics. *Annual review of biomedical engineering*, 11:109–134, 2009.
- [82] C. A. Taylor, T. A. Fonte, and J. K. Min. Computational fluid dynamics applied to cardiac computed tomography for noninvasive quantification of fractional flow reserve: Scientific basis. *Journal of the American College of Cardiology*, 61(22):2233–2241, 2013. ISSN 0735-1097. doi: <https://doi.org/10.1016/j.jacc.2012.11.083>. URL <https://www.sciencedirect.com/science/article/pii/S0735109713013041>.
- [83] C. Tesche and H. N. Gray. Machine learning and deep neural networks applications in coronary flow assessment: The case of computed tomography fractional flow reserve. *Journal of Thoracic Imaging*, 35, 2020. ISSN 0883-5993. URL https://journals.lww.com/thoracicimaging/Fulltext/2020/05001/Machine_Learning_and_Deep_Neural_Networks.11.aspx.
- [84] C. Tesche, C. N. De Cecco, M. H. Albrecht, T. M. Duguay, R. R. Bayer, S. E. Litwin, D. H. Steinberg, and U. J. Schoepf. Coronary ct angiography-derived fractional flow reserve. *Radiology*, 285(1):17–33, 2017.
- [85] G. Tetteh, V. Efremov, N. D. Forkert, M. Schneider, J. Kirschke, B. Weber, C. Zimmer, M. Piraud, and B. H. Menze. Deepvesselnet: Vessel segmentation, centerline prediction, and bifurcation detection in 3-d angiographic volumes. *Frontiers in Neuroscience*, 14, 2020. ISSN 1662-453X. doi: 10.3389/fnins.2020.592352. URL <https://www.frontiersin.org/article/10.3389/fnins.2020.592352>.
- [86] P. A. Tonino, B. De Bruyne, N. H. Pijls, U. Siebert, F. Ikeno, M. vant Veer, V. Klauss, G. Manoharan, T. Engstrøm, K. G. Oldroyd, et al. Fractional flow reserve versus angiography for guiding percutaneous coronary intervention. *New England Journal of Medicine*, 360(3):213–224, 2009.
- [87] G. Toth, M. Hamilos, S. Pyxaras, F. Mangiacapra, O. Nelis, F. de Vroey, L. Serafino,

- O. Muller, C. Mieghem, E. Wyffels, G. Heyndrickx, J. Bartunek, M. Vanderheyden, E. Barbato, W. Wijns, and B. Bruyne. Evolving concepts of angiogram: Fractional flow reserve discordances in 4000 coronary stenoses. *European heart journal*, 35, 03 2014. doi: 10.1093/eurheartj/ehu094.
- [88] G. G. Toth, B. Toth, N. P. Johnson, F. D. Vroey, L. D. Serafino, S. Pyxaras, D. Rusinaru, G. D. Gioia, M. Pellicano, E. Barbato, C. V. Mieghem, G. R. Heyndrickx, B. D. Bruyne, and W. Wijns. Revascularization decisions in patients with stable angina and intermediate lesions. *Circulation: Cardiovascular Interventions*, 7(6):751–759, 2014. doi: 10.1161/CIRCINTERVENTIONS.114.001608. URL <https://www.ahajournals.org/doi/abs/10.1161/CIRCINTERVENTIONS.114.001608>.
- [89] C. W. Tsao, A. W. Aday, Z. I. Almarzooq, A. Alonso, A. Z. Beaton, M. S. Bittencourt, A. K. Boehme, A. E. Buxton, A. P. Carson, Y. Commodore-Mensah, M. S. Elkind, K. R. Evenson, C. Eze-Nliam, J. F. Ferguson, G. Generoso, J. E. Ho, R. Kalani, S. S. Khan, B. M. Kissela, K. L. Knutson, D. A. Levine, T. T. Lewis, J. Liu, M. S. Loop, J. Ma, M. E. Mussolino, S. D. Navaneethan, A. M. Perak, R. Poudel, M. Rezk-Hanna, G. A. Roth, E. B. Schroeder, S. H. Shah, E. L. Thacker, L. B. VanWagner, S. S. Virani, J. H. Voecks, N.-Y. Wang, K. Yaffe, S. S. Martin, and null null. Heart disease and stroke statistics—2022 update: A report from the american heart association. *Circulation*, 145(8):e153–e639, 2022. doi: 10.1161/CIR.0000000000001052. URL <https://www.ahajournals.org/doi/abs/10.1161/CIR.0000000000001052>.
- [90] A. Uus, P. Liatsis, and R. Rajani. The influence of vessel segmentation threshold on the accuracy of patient-specific coronary blood flow simulations. pages 1–4, 10 2016. doi: 10.1109/MWSCAS.2016.7870126.
- [91] A. van der Giessen, M. Schaap, F. Gijsen, H. Groen, T. Walsum, N. Mollet, J. Dijkstra, F. van de Vosse, W. Niessen, P. Feyter, A. van der Steen, and J. Wentzel. 3d fusion of intravascular ultrasound and coronary computed tomography for in-vivo wall shear stress analysis: A feasibility study. *The international journal of cardiovascular imaging*, 26:781–96, 11 2009. doi: 10.1007/s10554-009-9546-y.
- [92] S. S. Virani, A. Alonso, H. J. Aparicio, E. J. Benjamin, M. S. Bittencourt, C. W. Callaway, A. P. Carson, A. M. Chamberlain, S. Cheng, F. N. Delling, M. S. Elkind, K. R. Evenson, J. F. Ferguson, D. K. Gupta, S. S. Khan, B. M. Kissela, K. L. Knutson, C. D. Lee, T. T. Lewis, J. Liu, M. S. Loop, P. L. Lutsey, J. Ma, J. Mackey, S. S. Martin, D. B. Matchar, M. E. Mussolino, S. D. Navaneethan, A. M. Perak, G. A. Roth, Z. Samad, G. M. Satou, E. B. Schroeder, S. H. Shah, C. M. Shay,

- A. Stokes, L. B. VanWagner, N.-Y. Wang, C. W. Tsao, and null null. Heart disease and stroke statistics—2021 update. *Circulation*, 143(8):e254–e743, 2021. doi: 10.1161/CIR.0000000000000950. URL <https://www.ahajournals.org/doi/abs/10.1161/CIR.0000000000000950>.
- [93] M. Vorre and J. Abdulla. Diagnostic accuracy and radiation dose of ct coronary angiography in atrial fibrillation: Systematic review and meta-analysis. *Radiology*, 267, 02 2013. doi: 10.1148/radiol.13121224.
- [94] A. Waljee and P. Higgins. Machine learning in medicine: A primer for physicians. *The American journal of gastroenterology*, 105:1224–6, 06 2010. doi: 10.1038/ajg.2010.173.
- [95] M. Wan, L. Ma, X. Zhao, S. Leng, J.-M. Zhang, R. S. Tan, and L. Zhong. Automatic segmentation of coronary artery lumen via anisotropic graph-cuts *. volume 2019, pages 4871–4874, 07 2019. doi: 10.1109/EMBC.2019.8856353.
- [96] Z.-Q. Wang, Y.-J. Zhou, Y.-X. Zhao, D.-M. Shi, Y. Liu, W. Liu, X.-L. Liu, and Y. Li. Diagnostic accuracy of a deep learning approach to calculate ffr from coronary ct angiography. *Journal of Geriatric Cardiology*, 16:42–48, 01 2019. doi: 10.11909/j.issn.1671-5411.2019.01.010.
- [97] O. Wink, A. F. Frangi, B. Verdonck, M. A. Viergever, and W. J. Niessen. 3d mra coronary axis determination using a minimum cost path approach. *Magn Reson Med*, 47(6):1169–1175, June 2002.
- [98] J. M. Wolterink, T. Leiner, and I. Išgum. Graph convolutional networks for coronary artery segmentation in cardiac ct angiography. In *International Workshop on Graph Learning in Medical Imaging*, pages 62–69. Springer, 2019.
- [99] J. M. Wolterink, R. W. van Hamersveld, M. A. Viergever, T. Leiner, and I. Išgum. Coronary artery centerline extraction in cardiac ct angiography using a cnn-based orientation classifier. *Medical Image Analysis*, 51:46–60, 2019. ISSN 1361-8415. doi: <https://doi.org/10.1016/j.media.2018.10.005>. URL <https://www.sciencedirect.com/science/article/pii/S1361841518308491>.
- [100] W. Wu, D.-R. Pan, N. Foin, S. Pang, P. Ye, N. Holm, X.-M. Ren, J. Luo, A. Nanjundappa, and S.-L. Chen. Noninvasive fractional flow reserve derived from coronary computed tomography angiography for identification of ischemic lesions: a systematic review and meta-analysis. *Scientific Reports*, 6(1):29409, Jul 2016. ISSN 2045-2322. doi: 10.1038/srep29409. URL <https://doi.org/10.1038/srep29409>.

- [101] R. Xiao, J. Yang, T. Li, and Y. Liu. Ridge-based automatic vascular centerline tracking in x-ray angiographic images. In J. Yang, F. Fang, and C. Sun, editors, *Intelligent Science and Intelligent Data Engineering*, pages 793–800, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-36669-7. doi: 10.1007/978-3-642-36669-7_96.
- [102] G. Yang, P. Kitslaar, M. Frenay, A. Broersen, M. J. Boogers, J. J. Bax, J. H. Reiber, and J. Dijkstra. Automatic centerline extraction of coronary arteries in coronary computed tomographic angiography. *The international journal of cardiovascular imaging*, 28(4):921–933, 2012.
- [103] G. Yang, P. Kitslaar, M. Frenay, A. Broersen, M. J. Boogers, J. J. Bax, J. H. C. Reiber, and J. Dijkstra. Automatic centerline extraction of coronary arteries in coronary computed tomographic angiography. *The International Journal of Cardiovascular Imaging*, 28(4):921–933, Apr 2012. ISSN 1573-0743. doi: 10.1007/s10554-011-9894-2. URL <https://doi.org/10.1007/s10554-011-9894-2>.
- [104] H. Yang, J. Chen, Y. Chi, X. Xie, and X. Hua. Discriminative coronary artery tracking via 3d cnn in cardiac ct angiography. In D. Shen, T. Liu, T. M. Peters, L. H. Staib, C. Essert, S. Zhou, P.-T. Yap, and A. Khan, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, pages 468–476, Cham, 2019. Springer International Publishing. ISBN 978-3-030-32245-8.
- [105] Y. Yang, A. Tannenbaum, D. Giddens, and A. Stillman. Automatic segmentation of coronary arteries using bayesian driven implicit surfaces. In *2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 189–192, 2007. doi: 10.1109/ISBI.2007.356820.
- [106] Y. Yin, M. Adel, and S. Bourennane. Retinal vessel segmentation using a probabilistic tracking method. *Pattern Recognition*, 45(4):1235–1244, 2012. ISSN 0031-3203. doi: <https://doi.org/10.1016/j.patcog.2011.09.019>. URL <https://www.sciencedirect.com/science/article/pii/S0031320311003980>.
- [107] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.
- [108] Y. Zheng, H. Tek, and G. Funka-Lea. Robust and accurate coronary artery centerline extraction in CTA by combining model-driven and data-driven approaches. *Med Image Comput Comput Assist Interv*, 16(Pt 3):74–81, 2013.
- [109] W. Zhu, N. Zeng, and N. Wang. Sensitivity, specificity, accuracy, associated confi-

- dence interval and roc analysis with practical sas (R) implementations. *NorthEast SAS users group, health care and life sciences*, 01 2010.
- [110] M. Zreik, N. Lessmann, R. W. van Hammersveld, J. M. Wolterink, M. Voskuil, M. A. Viergever, T. Leiner, and I. Išgum. Deep learning analysis of the myocardium in coronary ct angiography for identification of patients with functionally significant coronary artery stenosis. *Medical image analysis*, 44:72–85, 2018.
 - [111] M. Zreik, R. van Hammersveld, N. Khalili, J. Wolterink, M. Voskuil, M. Viergever, T. Leiner, and I. Isgum. Deep learning analysis of coronary arteries in cardiac ct angiography for detection of patients requiring invasive coronary angiography. *IEEE Transactions on Medical Imaging*, PP:1–1, 11 2019. doi: 10.1109/TMI.2019.2953054.
 - [112] M. Zreik, N. Hampe, T. Leiner, N. Khalili, J. M. Wolterink, M. Voskuil, M. A. Viergever, and I. Išgum. Combined analysis of coronary arteries and the left ventricular myocardium in cardiac ct angiography for detection of patients with functionally significant stenosis. In I. Išgum and B. A. Landman, editors, *Medical Imaging 2021: Image Processing*, volume 11596, pages 394 – 401. International Society for Optics and Photonics, SPIE, 2021. doi: 10.1117/12.2580847. URL <https://doi.org/10.1117/12.2580847>.



© Encyclopædia Britannica, Inc.

Figure A.1: Cross section of the human heart. Source: Encyclopædia Britannica, Inc. [22]

A | On coronary arteries and coronary artery disease

The main matter of this document consists in the analysis of computed tomography imagery of the heart and its components, with particular interest towards the coronary arteries. This appendix proposes a focused introduction to the heart topography and its vascular system, as well as coronary artery disease description, causes and figures. The terminology used throughout the document is introduced, and the relevance of coronary artery disease (CAD) is discussed.

A.1. The heart

The heart is the organ that serves as a pump to circulate the blood. In humans and other mammals and in birds, the heart is a four-chambered double pump that is the centre of the circulatory system. In humans it is situated at the center of the chest between the two lungs, slightly to the left, behind the breastbone; it rests on the diaphragm, the muscular partition between the chest and the abdominal cavity.

The heart consists of several layers of a tough muscular wall, the *myocardium*. A thin layer of tissue (*pericardium*) covers the outside, and another layer *endocardium* lines the inside. The heart cavity is divided down the middle into a right and a left heart, which in turn are subdivided into two chambers. The upper chamber is called an *atrium* (plural *atria*), and the lower chamber is called a *ventricle*. The two atria act as receiving chambers for blood entering the heart, and the more muscular ventricles pump the blood out of the heart.

The pumping of the heart is caused by alternating contractions and relaxations of the myocardium. These contractions are stimulated by electrical impulses from a natural pacemaker, the sinoatrial node, whose impulse causes the two atria to contract forcing blood into the ventricles. Contraction of the ventricles is controlled by impulses from the atrioventricular node. Following contraction, the ventricles relax, and pressure within them falls. Blood again flows into the atria, and the cardiac cycle starts again. The period of relaxation is called *diastole*. The period of contraction is called *systole*. Diastole is the longer of the two phases so that the heart can rest between contractions.

A.2. Coronary vasculature system

As any other organ, the heart vasculature is composed by veins and arteries. The vessels serving the muscles of the heart are called coronaries.

The coronary arteries are two blood vessels branching from the root of the aorta and carry oxygen-rich blood to the heart muscle. The coronary arteries wrap around the outside of the heart, and small branches dive into the heart muscle to bring it blood. Both arteries supply blood to the walls of both ventricles and to the partition between the chambers of the ventricles. The right main coronary artery supplies blood to the right atrium, while the left main supplies the left atrium.

The left main coronary (LCA, LMCA) artery supplies blood to the left side of the heart muscle, and it divides into branches.

The left anterior descending (LAD) artery branches off the left coronary artery and sup-

Coronary arteries of the heart

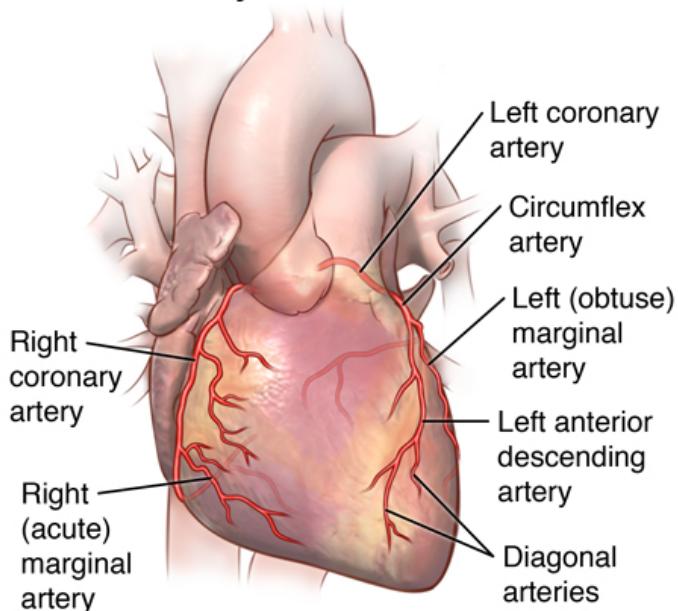


Figure A.2: Coronary arterial system. Source: Encyclopædia Britannica, Inc. [22]

plies blood to the front of the left side of the heart.

The left circumflex artery (LCX, CX) branches off the left coronary artery and encircles the heart muscle. This artery supplies blood to the outer side and back of the heart.

The LCA also gives off the left marginal artery (LMA). In 20-25% of individuals, the LCX contributes to the posterior interventricular artery (PIV).

The right coronary artery (RCA) supplies blood to the right ventricle, the right atrium, and the sinoatrial and atrioventricular nodes. The right coronary artery branches to form the right (acute) marginal artery (RMA) on the front and, in 80-85% of individuals, it also branches into the right posterior interventricular artery (PIV) on the back.

Smaller branches of the coronary arteries include: obtuse marginal branch, septal perforator, and diagonals (first, second, optional) branches.

The venous drainage of the heart is mostly through the coronary sinus – a large venous structure located on the posterior aspect of the heart. The cardiac veins drain into the coronary sinus, which in turn empties into the right atrium. There are also smaller cardiac veins which pass directly into the right atrium. The main tributaries of the coronary sinus are:

- Great cardiac vein – the largest tributary of the coronary sinus. It originates at the apex (the point) of the heart and ascends in the anterior interventricular groove. It

then curves to the left and continues onto the posterior surface of the heart. Here, it gradually enlarges to form the coronary sinus.

- Small cardiac vein – located on the anterior surface of the heart, in a groove between the right atrium and right ventricle. It travels within this groove onto the posterior surface of the heart, where it empties into the coronary sinus.
- Middle cardiac vein – begins at the apex of the heart and ascends in the posterior interventricular groove to empty into the coronary sinus.
- Posterior cardiac vein – located on the posterior surface of the left ventricle. It lies to the left of the middle cardiac vein and empties into the coronary sinus.

As it can be observed in figure A.3, coronary arteries and veins run side by side through most of the major branches, however veins all drains through the coronary sinus into the heart, while arteries spawn from the base of the aorta. In this document, the point where the coronary artery originates from the aorta is called *ostium* (from latin, plural *ostia*). Another feature of coronary arteries, and blood vessels in general, is of primary interest in this document: the *lumen* (from Latin, meaning "an opening"; plural *lumina*) is the inside space of a tubular structure, such as an artery or intestine. It can refer to the interior of a vessel, such as the central space in an artery, vein or capillary through which blood flows.

A.3. Coronary arteries disease and atherosclerosis

Atherosclerosis is a chronic disease caused by the deposition of fats, cholesterol, calcium, and other substances in the innermost layer of endothelium of the large and medium-sized arteries. Thick plaques that severely occlude an artery can significantly decrease the flow of blood to vascular beds in tissues served by the artery, thereby causing severe tissue damage. In addition, a disturbance to the endothelium may result in the formation of a blood clot (thrombus) at the site of a plaque, likewise obstructing the channel or breaking loose from the site and causing a catastrophic blockage elsewhere.

Atherosclerotic lesions frequently are found in the aorta and in large aortic branches. They are also prevalent in the coronary arteries, where the condition is called coronary heart disease. Coronary heart disease, also called coronary artery disease (CAD) or ischemic heart disease, is characterized by an inadequate supply of oxygen-rich blood to the myocardium because of narrowing (stenosis) or blocking of a coronary artery by atherosclerotic plaques. If the oxygen depletion is extreme, the effect may be a myocardial infarction (heart attack, with possible consequent death of a section of heart muscle).

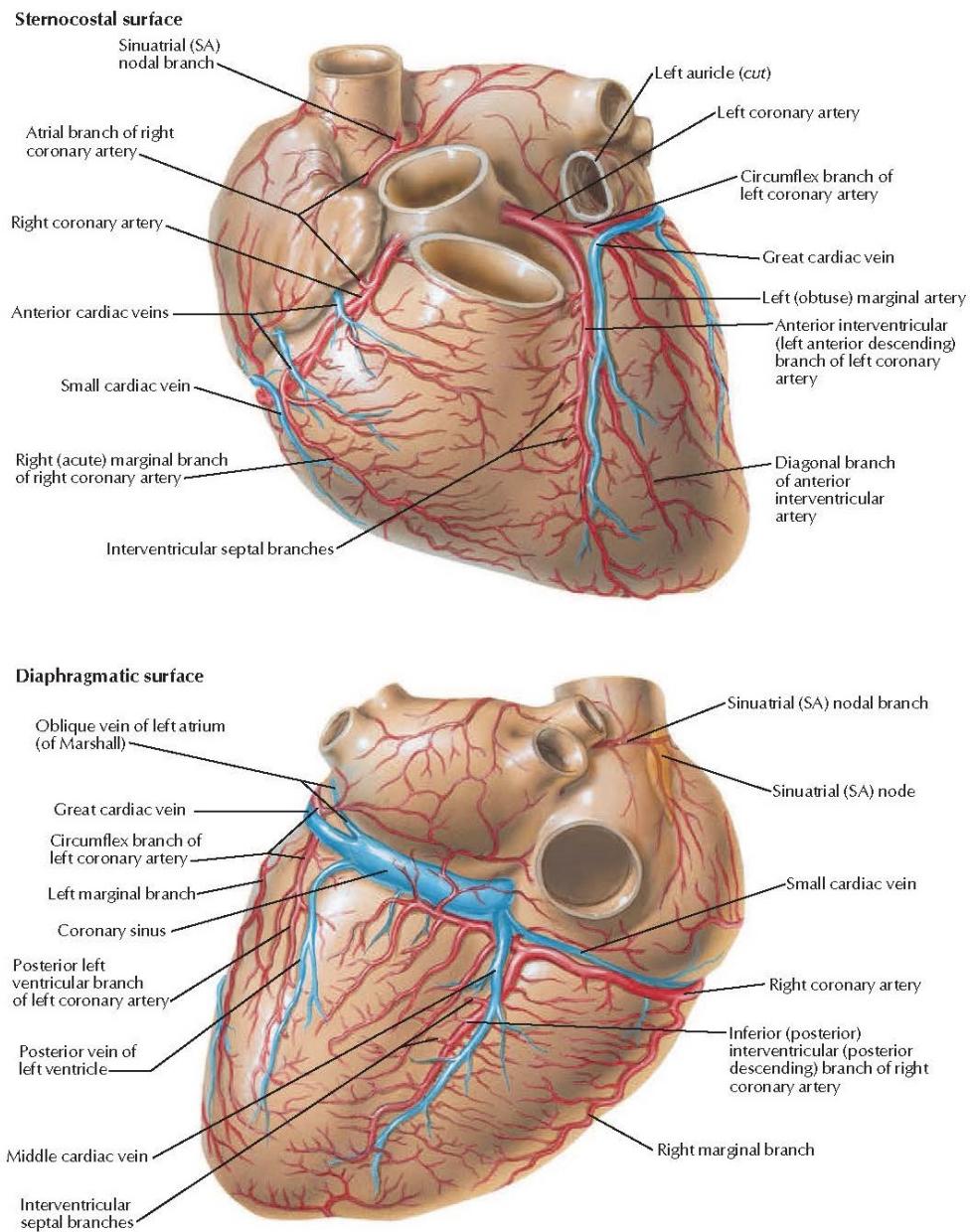


Figure A.3: Heart vascular system (Netter Basic Science). Source: Frank H. Netter, Atlas of Human Anatomy, Elsevier (2018), Page 321

If the deprivation is insufficient to cause infarction, the effect may be angina pectoris (pain or discomfort in the chest). Both conditions can be fatal because they can lead to heart failure or ventricular fibrillation. The latter, characterized by an uncontrolled and uncoordinated contraction of the ventricles (the lower chambers of the heart), can induce sudden death.

A family history of cardiovascular disease, smoking, stress, obesity, and high blood chole-

A | On coronary arteries and coronary artery disease

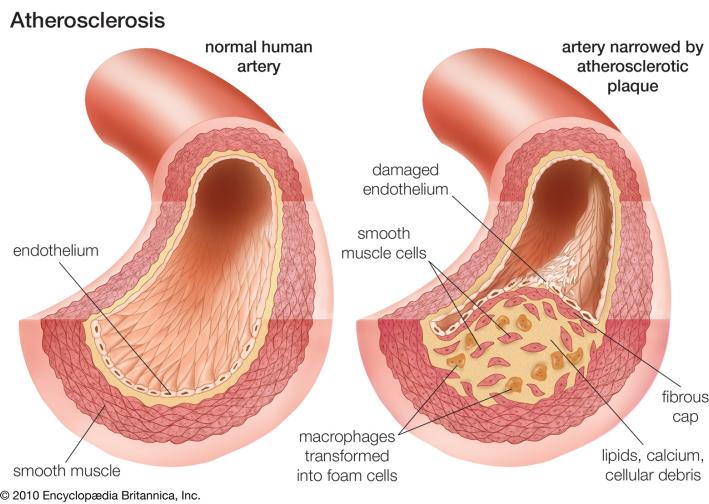


Figure A.4: Diagram of an atherosclerotic plaque. Source: Encyclopædia Britannica, Inc.

terol levels, particularly in association with LDLs, are among the factors that contribute to an increased risk of developing atherosclerosis. Men develop atherosclerosis more often than women, and individuals with diabetes mellitus have a significantly higher incidence of the disease.

Here are reported some statistics on coronary artery disease from the American Heart Association report from 2022 [89, 92].

Cardiovascular disease (CVD), listed as the underlying cause of death, accounted for 874'613 deaths in the United States in 2019. CVD claim more lives each year in the United States than all forms of cancer and chronic lower respiratory disease combined. Between 2017 and 2018, direct and indirect costs of total CVD were \$378.0 billion (\$226.2 billion in direct costs and \$151.8 billion in lost productivity/mortality). CVD accounted for approximately 19.05 million global deaths in 2020.

Heart Disease remains the No. 1 cause of death in the United States, according to 2019 data. In 2019 in the United States, coronary heart disease (CHD, dual name meaning CAD which also involves veins) was the leading cause (41.3%) of deaths attributable to CVD in the United States, followed by other CVD (17.3%), stroke (17.2%), high blood pressure (11.7%), heart failure (9.9%), diseases of the arteries (2.8%). CHD accounted for approximately 12.6% of deaths in the United States in 2018, causing 360,900 deaths. According to data from 2005 to 2014, the estimated annual incidence of heart attack in the United States was 605,000 new attacks and 200,000 recurrent attacks. Average age at the first heart attack was 65.6 years for males and 72.0 years for females. Approximately every 40 seconds, someone in the United States will have a myocardial infarction. From 2009 to 2019 in the United States, the annual death rate attributable to CHD declined

25.2% and the actual number of deaths declined 6.6%. Finally, the estimated direct and indirect cost of heart disease in 2017 to 2018 (average annual) was \$228.7 billion in the United States.

A.4. Common revascularisation procedures

In the case of occluded coronary arteries, the lives of countless cardiac patients have been saved by coronary bypass surgery, in which sections of blood vessels from other parts of the body are used to route blood flow around the obstructions. Some occlusions can be opened by balloon angioplasty (figure A.9), in which a catheter is inserted to the site of obstruction and a balloon is inflated in order to dilate the artery and flatten the plaque deposits. Passages opened in this way frequently reclose over time, but the chances of this occurring can be reduced significantly by the insertion of expandable wire-mesh stents as part of the angioplasty procedure. Some stents are “drug-eluting,” that is, coated with a drug that inhibits the kind of cell growth that leads to reclosure.

A| On coronary arteries and coronary artery disease

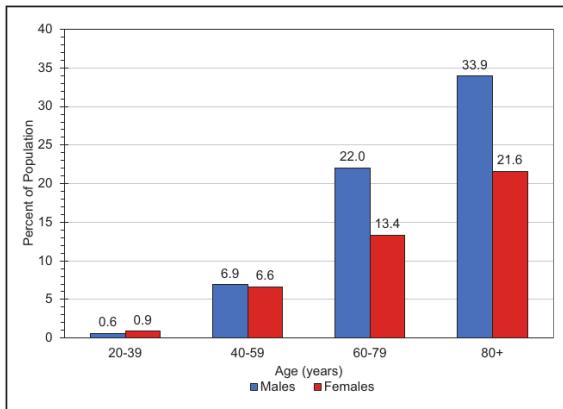


Figure A.5: Prevalence of coronary heart disease, by age and sex, United States, 2015–2018 (NHANES, National Health and Nutrition Examination Survey). Source: AHA [89].

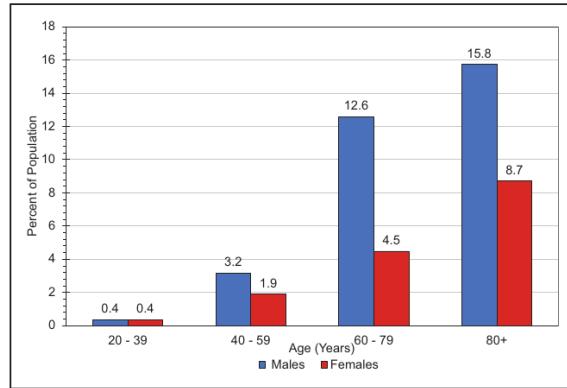


Figure A.6: Prevalence of myocardial infarction, by age and sex, United States, 2015–2018 (NHANES). Source: AHA [89].

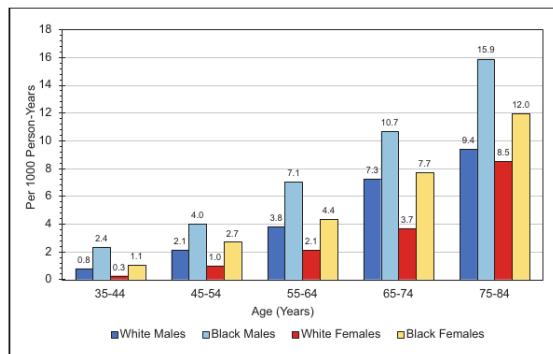
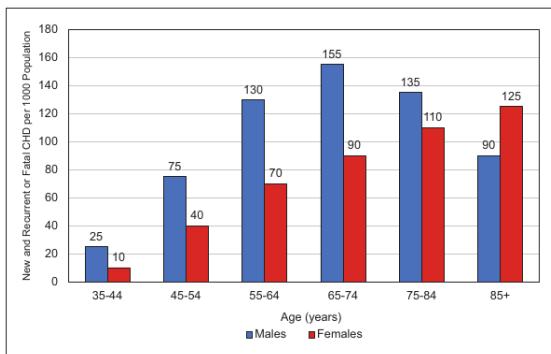


Figure A.7: Left: Annual number of US adults per 1000 having diagnosed heart attack or fatal coronary heart disease, by age and sex, 2005–2014 (ARIC Surveillance, and CHS). Right: Incidence of MI, by age, sex, and race, United States, 2005–2014 (ARIC Surveillance). Source: AHA [89].

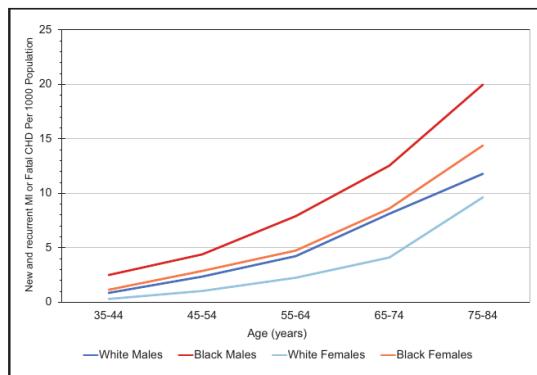


Figure A.8: Incidence of heart attack or fatal coronary heart disease, by age, sex, and race, United States, 2005–2014 (ARIC Surveillance). Source: AHA [89].

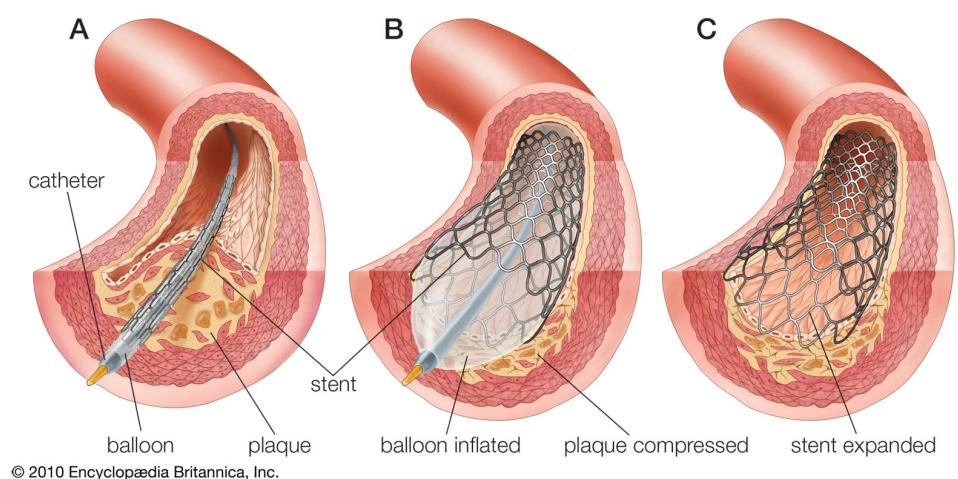


Figure A.9: Balloon angioplasty and stent insertion.

- In a coronary artery where blood flow is obstructed by the growth of atherosclerotic plaque, the point of obstruction is reached by a cardiac catheter encased in an inflatable balloon and wire-mesh stent.
- The balloon is inflated, thus expanding the stent, dilating the artery, and compressing the plaque.
- The balloon is deflated and withdrawn with the catheter, leaving the stent expanded against the arterial wall.

Source: Encyclopædia Britannica, Inc.

B | On CCTA images

Non-invasive cardiovascular imaging has traditionally been stratified into two distinct types: anatomic imaging and functional imaging. The chief aim of anatomic imaging is the identification of atherosclerosis, while functional imaging provides information relating to the hemodynamic significance of that atherosclerosis.

Anatomic imaging modalities include multi-detector computed tomography (MDCT, or simply CT), electron beam computed tomography (EBCT) and magnetic resonance angiography (MRA), while functional imaging modalities include single photon emission computed tomography (SPECT), myocardial perfusion imaging (MPI), stress echocardiography (SE), cardiac magnetic resonance imaging (CMR), and positron emission tomography (PET) perfusion imaging. Currently, the assessment of functionally significant coronary heart disease (CHD) has relied primarily upon nuclear MPI or stress echocardiography (SE). Both MPI and SE have demonstrated robust ability to diagnose significant CHD, as well as the capability to prognosticate future outcomes.

CTA (computed tomography applied to coronaries analysis and angiography; also "cardiac/coronary CT angiography", CCTA) is a non-invasive technique that allows, next to the assessment of the coronary lumen, the evaluation of the presence, extent, and type (non-calcified or calcified) of coronary plaque (Leber et al., 2006). Such non-invasive, comprehensive plaque assessment may be relevant for improving risk stratification when combined with current risk measures: the severity of stenosis and the amount of calcium. Typically, non-contrast CT imaging exploits the density of tissues to generate an image, such that different densities have different attenuation values: soft tissues, calcium, fat, and air can be easily distinguished, and thus allows to estimate the amount of calcium present in the coronary arteries [38]. In comparison, contrast-enhanced CCTA, which is acquired after the injection of a contrast agent into the blood flow, can provide excellent visualization of cardiac chambers, vessels and coronaries, and has been shown to be effective in detecting non-calcified coronary plaques [13]. Considering the task at hand, particular interest falls on contrast-enhanced CCTA.

As in any imaging technique, the more pixels – voxels – are available per unit volume,

the more details and information the image will hold, allowing for more precise geometric reconstruction of the interesting structure (such as coronary vessels). Within the scope of automatic coronary artery tree extraction, this condition is particularly important to obtain correct reconstruction with less artifacts.

A disadvantage of CTA is that the current imaging protocols are associated with a higher radiation dose exposure than CCA. The radiation dose the patient is subject to during the imaging phase depends on the applicable protocols, but also on the modality of acquisition of such images. Usually, prospective electrocardiography triggering requires less radiation dose than retrospective electrocardiography gating [93] but theoretically provides less cardiac phases for interpretation, which also translates in worse image quality. However a study from Menke et al. [52] in 2013 proved that in patients with coronary artery disease and without tachyarrhythmia, prospectively triggered CCTA provides image quality and diagnostic accuracy comparable with retrospectively gated CCTA, but at a much lower radiation dose.

In conclusion, the Biomedical Imaging Group Rotterdam ¹ described in details in [72] the characteristics of the CCTA images used nowadays by all the researchers as a standardised set of data on which to build and evaluate their algorithms. CCTA images can vary within tolerances, however the same basic characteristics should be sought as the ones described in the referenced paper. Following is reported the standard cardiac CT dataset composition.

The computed tomography angiography (CTA) data was acquired in the Erasmus MC, University Medical Center Rotterdam, The Netherlands. Thirty-two datasets were randomly selected from a series of patients who underwent a cardiac CTA examination between June 2005 and June 2006. Twenty datasets were acquired with a 64-slice CT scanner and 12 datasets with a dual-source CT scanner (Sensation 64 and Somatom Definition, Siemens Medical Solutions, Forchheim, Germany). A tube voltage of 120 kV was used for both scanners. All datasets were acquired with ECG-pulsing. The maximum current (625 mA for the dual-source scanner and 900 mA for the 64-slice scanner) was used in the window from 25% to 70% of the R–R interval and outside this window the tube current was reduced to 20% of the maximum current. Both scanners operated with a detector width of 0.6 mm. The image data was acquired with a table feed of 3.8 mm per rotation (64-slice datasets) or 3.8 mm to 10 mm, individually adapted to the patient’s heart rate (dual-source datasets). Diastolic reconstructions were used, with reconstruction intervals varying from 250 ms to 400 ms before the R-peak. Three datasets were reconstructed using a sharp (B46f) kernel, all others were reconstructed using a medium-to-smooth (B30f)

¹Link to their home page.

kernel. The mean voxel size of the datasets is 0.32x0.32x0.4 mm³.

C | On supervised machine learning

Machine learning (ML) is a field of computer science that uses computer algorithms to identify patterns in large amounts of data, and can be used to perform various tasks based on the data [1, 31]. Machine learning algorithms typically calibrate a model by using a copious amount of test data during the *training* phase, in order to perform data-driven predictions or decisions after it has been trained. In recent years, machine learning techniques have emerged as highly effective methods for prediction and decision-making in a multitude of disciplines, including internet search engines, customized advertising, natural language processing, finance trending, computer vision, robotics [18, 94] and healthcare [59]. For example, the latest advancements in self-driving vehicles all make extensive use of machine learning and artificial intelligence techniques.

A more formal definition would assert that "*a computer program is said to learn from experience with respect to some task and some performance measure if its performance on the task improves with experience.*" This method arises at the intersection of statistics, which seeks to learn relationships from data, and computer science, with its emphasis on efficient computing algorithms. This union between mathematics and computer science is driven by the unique computational challenges of building statistical models from massive data sets, which can include up to billions or trillions of data points.

Supervised learning, classification, regression

The types of learning used by computers are conveniently sub-classified into categories such as *supervised* learning and *unsupervised* learning. In addition, another division can be useful when considering how machine learning might inform the practice of medicine: distinguishing learning those tasks that physicians can already do well opposed to learning those where physicians have had only limited success [18] – such as non-invasive estimation the functional severity of a coronary stenosis. Since the scope of this thesis is focused in solving a supervised learning problem, here are reported some valuable additional information about this class of methods.

Supervised learning is defined by its use of *labeled data*¹ to train algorithms to classify data or predict outcomes accurately. As input data is fed into the model, it adjusts its free parameters until the model has been fitted appropriately. Training data includes inputs (raw data) and correct outputs (labels), which allows the model to learn over time. The algorithm measures its accuracy through a *loss function* – a way to measure performance – adjusting the free parameters of the model until the error between the real output and predicted output has been sufficiently minimized.

Another sub-classification is possible, and it is in our interest to understand the difference between the two approaches:

Classification uses an algorithm to accurately assign test data into specific classes or *categories*. It recognizes specific entities within the dataset and attempts to draw some conclusions on how those entities should be labeled or defined. In our case, the problem can be formulated as a classification problem as follows: "given a series of CCTA images and their labels (useful information such as geometry and FFR values), the model is trained to understand if a patient has at least one stenosis with FFR_{CT} below or above a threshold (commonly 0.8)". The most common classification algorithms are "linear classifiers", "support vector machines" (SVM), "decision trees", "k-nearest neighbor", "neural networks" (NN) and "random forest".

Regression is used to understand the *functional relationship* between dependent and independent variables. It is commonly used to make projections, such as for sales revenue for a given business or weather forecasts. In our case, the problem can be formulated as a regression problem as follows: "given a series of CCTA images and their labels, the model is trained to estimate the exact value of the FFR index, either globally (given a CCTA image, the lowest value of the FFR_{CT} is found) or locally (FFR_{CT} is determined along the coronary artery tree, either in a *per-artery-segment* basis or a *whole-centerline-points* basis)". The most common regression algorithms are "linear regression", "logistical regression", "polynomial regression" and again "neural networks".

It is important to notice that the amount of data and the quality and precision of the labels needed for training varies with respect to one method or the other: for classification purposes, a simple binary classification of the image could be enough, while for regression more accurate labels are needed, such as, for example, the punctual values of the invasive FFR measurements along the coronary artery tree or some of its branches. This trade-off is mostly up to the data providers: a heavy-duty labeling activity for training is repaid

¹A datum (piece of information that a computer can understand) is said to be labeled if, in addition to the raw datum, also some other information is provided, essential for interpreting the datum. Once the model has been trained, only raw data will be used to make predictions, however the model needs the additional information in order to train. It can be thought as solving an exercise or a problem with the solution at hand, versus (once you are properly trained) solving it by your own.

with more spatially-specific FFR_{CT} values during implementation and generally more accurate results, while simpler binary labels (such as "in this image, FFR is above/below a certain threshold") allow only up to a simple classification of the image.

Deep learning

A neural networks processes training data by mimicking the biological connections of the human brain through layers of nodes – also called neurons. Each node is made up of inputs (one or many), weights, a bias, an activation function, and an output. If that output value exceeds a given threshold, it "fires" or activates the node, passing data to the next layer in the network. Neural networks learn this mapping function through supervised learning, adjusting based on the loss function. At each learning cycle, the parameters of the network are re-calibrated so that the prediction error would be better than before. The calibration process has a better chance at being successful the more training data are provided.

Neural networks are primarily leveraged for deep learning algorithms. In deep learning, neural networks with three or more layers of neurons are used. While a neural network with a single layer can still make approximate predictions, additional "*hidden*" layers can help to optimize and refine the model for better accuracy. Each hidden layer builds upon the previous layer to refine and optimize the prediction quality. Deep learning also eliminates some of data pre-processing that is typically involved with machine learning. These algorithms can ingest and process unstructured data, like images, and it automates feature extraction, like coronary artery tree extraction, removing some of the dependency on human experts and manual labor. As more layers exist, more parameters are needed to describe the network, so more data are needed to tune the parameters to obtain a sufficiently low prediction error on never-seen-before inputs.

Supervised learning in daily clinical practice

It is important to note that seemingly large enough medical datasets and adequate learning algorithms have been available for many decades, and yet, although there are thousands of papers applying machine learning algorithms to medical data, very few have contributed meaningfully to clinical care in every day practices. This lack of impact stands in stark contrast to the enormous relevance of machine learning to many other industries.

Perhaps some of the few most common example of supervised learning seen by a cardiologist is the automated interpretation of the ECG, where pattern recognition is performed to select from a limited set of diagnoses (a classification task). In radiology, automated detection of a lung nodule from a chest x-ray would also represent supervised learning. In both these cases, the computer is approximating what a trained physician is already capable of doing with high accuracy, but much faster [18]. The healthcare industry has

already benefited greatly from these early examples of deep learning capabilities ever since the digitization of hospital records and images. Image recognition applications can support medical imaging specialists and radiologists, helping them analyze and assess more images in less time.

Slower training, faster predictions

Deep learning requires a tremendous amount of computing power during the training phase. High performance graphical processing units (GPUs) are ideal because they can handle a large volume of calculations in multiple cores with copious memory available. It has to be noted though that the only resource-intensive part of a ML approach is the training. Once a deep neural network is properly trained, the prediction on new, never seen before data is performed in a matter of seconds [96]. This is quite the opposite of fluid dynamic simulation techniques, where heavy-duty computations are performed in the prediction phase, rather than in the model building phase [43, 61]. This has some remarkable consequences: while a trained neural network can run on any hardware with no particular expectations in computing power, fluid dynamic simulations require to have access to high-performance devices in everyday practices.

The two methods are then in opposition: one requires a lot of time (even days) and resources for the training phase, but absolutely negligible resources (milliseconds) for the prediction phase; the other requires fewer resources in building the model, but significantly higher time (from tens of minutes to even hours) and computing resources in the prediction phase.

From a clinical practice point of view, a faster prediction should be preferred assuming comparable prediction performance. Training will be performed offline² only once on dedicated hardware before and/or separately from the actual implementation, which can take place on an everyday laptop computer.

Some additional resources can be found at <https://playground.tensorflow.org/>.

²*Offline* generally means that a procedure is performed non-concurrently with – separately from – the main process. The training and implementation phases do not share time and computing resources.

D | On coronary artery tree extraction from other imaging sources

The lumen segmentation, centerline extraction and coronary artery disease assessment tasks are not exclusively performed starting from CCTA images, but also other medical imaging sources and the recent literature is rich of examples.

Intravascular ultrasound (IVUS) is commonly considered to be the method of choice in assessing atherosclerotic plaque burden [56] and plaque characteristics [10]. It has been used extensively for tree reconstruction, too, oftentimes in combination with CT images. However, IVUS is an invasive method which has risks, it cannot be used to study the entire arterial tree and it is not indicated for asymptomatic individuals. Being this method invasive and being the focus of this document on the FFR index, it would be more efficient to measure pressure values directly while inside the arteries during the IVUS procedure rather than trying to estimate the FFR index from the IVUS images.

This document discusses coronary artery tree reconstructions performed starting by CCTA images, as the benefits of using this non-invasive imaging technique are evident and commonly recognised both from a logistical and economical standpoint. Fusion methods – such as CCTA-IVUS or CCTA plus optical coherence tomography – will not be considered since many studies [74, 79] show how fusion methods are no more a strict necessity, although preponderant in clinical practice [4, 9, 10, 91].

Also, since the main focus of this document lays in analysing techniques exploiting CCTA tomographic images, it is necessary to disregard a quite consistent portion of literature describing 3-dimensional quantitative coronary analysis methods (commonly referred to as 3D-QCA), which usually exploit two or multiple classic angiographic bi-dimensional images, biplane orthogonal angiography, or multi-plane angiography. Nonetheless, two remarkable examples are reported.

Schuurbiers et al. [74] validated an industrial implementation of 3D-QCA in the software

suite CAAS® from Pie Medical Imaging®. A more recent work by Siogkas et al. [79] in 2015 proposed a coronary reconstruction method based on angiographic images, and validated their 3D-QCA approach against a hybrid IVUS and angiography reconstruction method, showing how stand-alone, non-invasive angiographic analysis can be as valid as fusion methods.

The studies reported in this appendix not only show some interesting alternatives, but also show that standalone, non-invasive angiographic imagery potentially retain enough intrinsic information allowing the assessment of the severity of various causes of coronary artery disease.

E | On the Rotterdam Coronary Artery Algorithm Evaluation Framework

Efficiently obtaining a reliable coronary artery centerline from computed tomography angiography data is a very relevant task in modern clinical practice. Whereas numerous methods have been presented for this purpose, no standardized evaluation methodology has been published until 2009 to reliably evaluate and compare the performance of the existing or newly developed coronary artery centerline extraction algorithms.

A paper from Schaap et al. from the Biomedical Imaging Group Rotterdam [72] describes a standardized evaluation methodology and reference database for the quantitative evaluation of coronary artery centerline extraction algorithms - commonly referred to by researchers as the "Rotterdam Coronary Artery Algorithm Evaluation Framework".

The contribution of this work is fourfold:

1. a method is described to create a consensus centerline with multiple observers (reference centerlines and radii manual annotation from a group of radiologists);
2. well-defined measures are presented for the evaluation of coronary artery centerline extraction algorithms;
3. a database containing 32 cardiac CTA (CCTA) datasets with corresponding reference standard is described and made available;
4. 13 coronary artery centerline extraction algorithms, implemented by different research groups as part of a segmentation challenge workshop (coronary artery tracking challenge, CAT08) at the 2008 Medical Image Computing and Computer-Assisted Intervention (MICCAI) conference, are quantitatively evaluated and compared.

The presented evaluation framework have been made available to the medical imaging community for benchmarking existing or newly developed coronary centerline extraction algorithms.

E.1. Training set, test set and reference standard

The modality of acquisition of the CCTA images is discussed at the end of Appendix B and more in details in [72]; here the composition of the dataset is reported.

To ensure representative training and test sets, the image quality of - and presence of calcium in - each dataset was visually assessed by a radiologist with three years experience in cardiac CT. Image quality was scored as poor (defined as presence of image-degrading artifacts and evaluation only possible with low confidence), moderate (presence of artifacts but evaluation possible with moderate confidence) or good (absence of any image-degrading artifacts related to motion and noise). Presence of calcium was scored as absent, modest or severe. Based on these scorings the data was distributed equally over a group of 8 and a group of 24 datasets. The patient and scan parameters were assessed by the radiologist to be representative for clinical practice. The first group of 8 datasets can be used for training and the other 24 datasets are used for performance assessment of the algorithms. All the 32 CCTA datasets and the corresponding reference standard centerlines for the training data are made publicly available on the [website](#) of the institution or by directly contacting the authors.

The centerline of a coronary artery in a CTA scan as the curve that passes through the center of gravity of the lumen in each cross-section. The start point of a centerline is defined as the center of the coronary ostium (the point where the coronary artery originates from the aorta, see Appendix A), and the end point as the most distal point where the artery is still distinguishable from the background.

The observers also specified the radius of the lumen, where the radius was chosen such that the enclosed area of the annotated circle matched the area of the lumen.

Four vessels are selected for annotation in all 32 datasets, yielding 128 selected vessels. The first three vessels were always the right coronary artery (RCA), left anterior descending artery (LAD), and left circumflex artery (LCX). The fourth vessel was selected from the large side-branches of these main coronary arteries and varies from image to image, including the first diagonal branch, second diagonal branch, optional diagonal coronary artery, first obtuse marginal branch, posterior descending artery, and acute marginal artery (see figure E.1).

E.2. Standard evaluation framework

All the evaluation measures are based on a point-to-point correspondence between the reference standard and the evaluated centerline. Before the point-to-correspondence is

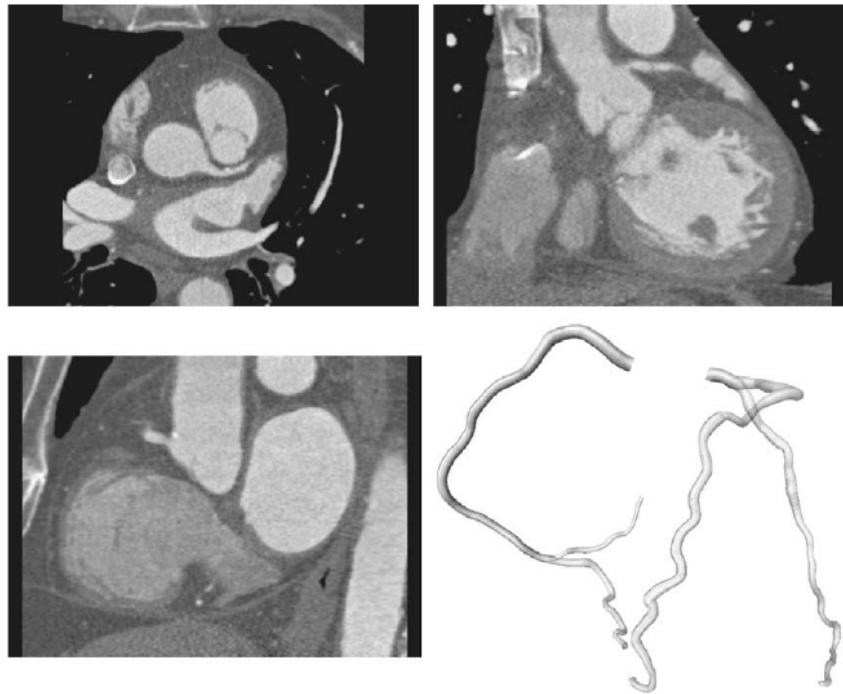


Figure E.1: An example of the data with corresponding reference standard. Top-left: axial view of data. Top-right: coronal view. Bottom-left: sagittal view. Bottom-right: a 3D rendering of the reference standard. Source: Schaap et al., 2008 [72].

determined the evaluated centerlines are first sampled equidistantly using a sampling distance of 0.03mm, enabling an accurate comparison. The evaluated centerline is then clipped with a disc that is positioned at the start of the reference standard centerline; The radius of the disc is twice the annotated vessel radius and the disc normal is the tangential direction at the beginning of the reference standard centerline; every point before the first intersection of an extracted centerline and this disc is not taken into account during evaluation.

The correspondence is then determined by finding the minimum of the sum of the Euclidean lengths of all point-point connections that are connecting the two centerlines (reference and extracted) over all valid correspondences. The valid correspondences and their conditions are precisely defined in [72], and they guarantee that each point of the reference standard is connected to at least one point of the extracted centerline and vice-versa. Dijkstra's graph search algorithm is used on a matrix with connection lengths to determine the minimal Euclidean length correspondence.

Once the point-to-point correspondence has been completely defined, the correspondence is used to define many general purpose evaluation measures. The authors define the following measures (also illustrated in figure E.2), reported here for reference purpose and exhaustively defined in the original paper [72]:

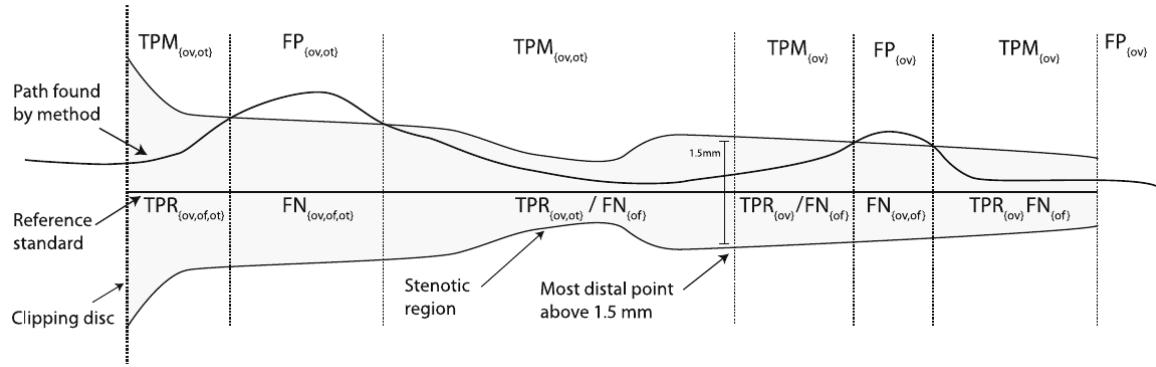


Figure E.2: A geometric illustration of the terms used in the evaluation measures. The reference standard with annotated radius is depicted in gray. The terms on top of the figure are assigned to points on the centerline found by the evaluated method. The terms below the reference standard line are assigned to points on the reference standard. Source: Schaap et al., 2008 [72].

- **Overlap (OV)** represents the ability to track the complete vessel annotated by the human observers.
- **Overlap until first error (OF)** determines how much of a coronary artery has been extracted before making an error.
- **Overlap with the clinically relevant part of the vessel (OT)** gives an indication of how well the method is able to track the section of the vessel that is assumed to be clinically relevant.
- **Average inside (AI)** is the average distance of all the connections between the reference standard and the automatic centerline given that the connections have a length smaller than the annotated radius at the connected reference point. The measure represents the accuracy of centerline extraction, provided that the evaluated centerline is inside the vessel.

With these measures we discern between extraction capability (OV, OF, OT) and extraction accuracy (AI). In order to discern between tracking ability and tracking accuracy, the accuracy is evaluated only within sections where tracking succeeded.

Scores are created from the extracted centerline evaluation measures by relating the measures to the three observers' performances. Each of the evaluation measures is related to the performance of the observers by a relative score such that a score of 100 points implies that the result of the method is perfect, 50 points implies that the performance of the method is similar to the performance of the observers, and 0 points implies a complete

failure.

The centerline tracking algorithms can then be ranked based on the obtained metrics: 4 metrics for each of the 94 tested vessels.

List of Figures

1.1	Illustration of a catheter-guidewire pressure measuring system. The catheter enters from the aorta and the guidewire slides towards distal regions of the coronary vessel. Source: Kobayashi et al., 2015 [44].	2
2.1	HeartFlow® FFR _{CT} estimation pipeline based on CFD. Source: heartflow.com	11
2.2	Schematic representation of the value of CPR in imaging complex vascular anatomy, both for humans and artificial intelligence. (A) Axial sections show only segments of the artery in cross section. (B) Multiplanar reconstruction in the orthogonal plane displays only various oblique cross sections of the artery. (C) CPR generated by centerline tracking captures and displays the entire artery. Source: radiologykey.com	16
2.3	Overview of the method proposed by Wolternik et al. (3D-CNN tracker) [99]. At location x , an isotropic 3D patch P is extracted and used as input to a convolutional neural network (CNN). This CNN simultaneously determines a probability distribution over a discrete set of directions on a sphere (here shown as a blue circle), and an estimate r of the radius of the vessel (as defined in Appendix E and [72]). Source: Wolternik et al., 2018 [99].	17
2.4	Detailed illustration of the DCAT method proposed by Yang et al. The backbone is a 3D CNN combined with first 4 convolutional layers used Wolternik's work [99]. Dilation convolutional layers are used and dilation rate in layer 2 and 3 is 2 and 4, respectively. The tracker has three 3D convolutional layers and two outputs: the radius estimate r and direction probability distribution T . The discriminator has three 3D convolutional layers and outputs a confidence p . Source: Yang et al., 2019 [104].	18
2.5	Overview of the AuCoTrack pipeline. Source: Salahuddin et al., 2021 [69].	19

2.6	Overview of the 3D-CNN tracker + GCN method proposed by Wolternik et al. [98] Given a coronary centerline C , rays are cast equiangularly and orthogonally to the centerline at each point c_i . Each ray angle corresponds to a vertex v in the luminal surface mesh of which the exact spatial location is determined by c_i , an angle ϕ_v , and the distance r_v . Image information along x_v is used by a GCN that combines this information with that of neighboring vertices to predict r_v . Source: Wolternik et al., 2019 [98].	21
2.7	Pipeline of joint coronary centerline extraction and lumen segmentation method (3D-CNN tracker + GCN) proposed by Gao et al. Source: Gao et al., 2021 [26].	22
2.8	Pipeline of the method proposed by Zreik et al. In a CCTA scan, the centerlines of the coronary arteries are extracted and used to reconstruct straightened multi-planar reformatted (MPR) images of the coronary arteries. Then, an unsupervised analysis is performed, where the MPR volume of a complete artery is encoded into a fixed number of features (see adjacent image). Then, the final extracted encodings are employed in a supervised fashion to classify arteries according to the need of further invasive evaluation to establish the need of intervention, using a support vector machine (SVM) classifier. Source: Zreik et al., 2018 [111].	25
2.9	The MPR volume of a complete artery is encoded into a fixed number of features (or encodings) using two disjoint convolutional autoencoders, applied sequentially: a 3D variational convolutional autoencoder (3D-VCAE) that spatially encodes local sub-volumes of the coronary artery, and a 1D convolutional autoencoder (1D-CAE) that sequentially encodes the encodings of the complete artery. Source: Zreik et al., 2018 [111].	25
2.10	Illustration of the pipeline for a patient with disease in the left anterior descending coronary artery. (a) CT imaging data. (b) Segmented geometry and FFRCT analysis. (c) FFRCT results in segmented and synthetic vasculature. (d) Illustration of the coupling loop, demonstrating quantities exchanged between the coronary model (left) and the myocardium model (right) at coupling iteration k. (e) Hyperemic MBF for coupled model. (f) Comparison of simulated (right) and PET (left) hyperaemic perfusion maps. Source: Papamanolis et al., 2020 [64].	27
2.11	Overview of the end-to-end pipeline to estimate the FFR _{CT} index.	28

3.1	Representation of the tracker and its modules (boxes) and interfaces (arrows). Any number of additional modules could be added to the tracker and connected to any other component through interfaces. Moreover, any network can be plugged in in the middle of the "visual cortex", provided that "input processing" and "output interpretation" are extended accordingly. For what concerns the proposed networks, any network can be plugged in and out at will without needing any other intervention on the tracker.	32
3.2	An example of 3D kernel and its movement across a portion ov volume. Source: towardsdatascience.com.	34
3.3	An example of 2D kernel dilation, which increases the receptive field. Source: Yu, Koltun, 2016 [107]. Another source: github.com/vdumoulin/conv_arithmetic/blob/master/gif/dilation.gif	34
3.4	An example of a 3D convolution operation. (a) 3D convolution of a input volume/feature map with a filter. (b) Generation of the i_{th} feature map in the next layer. Source: Liu et al, 2017, DOI: https://doi.org/10.1371/journal.pone.0185844	35
3.5	I/O scheme of Wolternik's CNN [99]. Input represented in 2D for clarity, the patch P is 3D, and a square pixel in the image represents a cubic voxel in the CCTA. Source: Wolternik et al., 2018 [99].	37
3.6	The set of possible directions distributed on a sphere (also called "shell") and the probability distribution on the shell. Source: Wolternik et al., 2018 [99].	37
3.7	Visual representation of the structure of the midsplit-4-FD network. The estimated direction classes are fed into the last layers of the radious regression branch of the network. This network amounts to $\sim 270'350$ trainable parameters.	43
3.8	a) The original centerlines unstructured point cloud. b) The two arterial trees are automatically classified correctly into two disjointed sets through a DBSCAN-inspired algorithm. To test the algorithm, a collage of 3 extraction process was created so that the resulting arterial trees are almost in contact, thus representing the hardest possible (yet unlikely) condition for the algorithm. The algorithm is able to separate the two arterial trees by considering 6 nearest neighbours.	54
3.9	Clustered versors in the 3D space of a point in the centerline. It is clear that the versors point towards two directions, which means that this point is not eligible to be a candidate endpoint.	56

3.10	Green endpoints found by the proposed procedures. Many points are found near the extremis, it is sufficient to classify them once again (this time the task is far easier) and taking the mean for each cluster.	56
4.1	Wolternik's network, standard training strategy. Distributions of R_{err} (a), $R_{err,abs}$ (b) and $R_{err,rel}$ (c). Lowermost: correlation between R_{err} and D. For every infographic shown in this section extensive description of the graphs is reported in the main text.	62
4.2	Wolternik's network [99], standard training strategy. Dependence of a) R_{est} , b) R_{err} and $R_{err,abs}$, c) $R_{err,rel}$ on R_{ref}	63
4.3	Wolternik's network [99], standard training strategy. Distribution and correlations of the angular error on the estimated and reference directions. The contents of this image are discussed in the main text.	64
4.4	Wolternik's network [99], standard training strategy. Distribution and correlations of the distance between $P_{landing}$ and the centerline.	65
4.5	Wolternik's network [99], standard training strategy. Distribution and correlations of the distance between $P_{landing}$ and the intended (reference) landing point on the centerline the network was trained to find (cfr section 3.2.2).	66
4.6	Wolternik's network [99], standard training strategy. Distribution and correlations of the movement termination metric computed through the normalized entropy $H(p(D P))$ of the posterior probability distribution on the direction classes.	67
4.7	Seeds used in the global test suite (green and orange) against the reference centerlines (blue).	68
4.8	Centerlines extracted from 5 seeds with the basic tracker.	72
4.9	Centerlines extracted from 5 seeds with Wolternik's tracker.	73
4.10	Centerlines tracking referred to the reference vessels' centerlines and radii, 5 seeds with Wolternik's tracker [99]. The vessel's reference centerline is straightened on the $y = 0$ axis of the figure, while the shadowed patch represents the inside of the vessel [72]. Tracking seems wiggly just because of the method used to compute connections between reference and extracted centerlines.	73
4.11	Centerlines extracted from 2 seeds with Wolternik's tracker [99]. Two of the four reference vessels are completely ignored.	74
4.12	Centerlines tracking referred to the reference vessels' centerlines and radii, 2 seeds with Wolternik's tracker [99]. From this graphs it is evident that two vessels were not correctly mapped.	74

4.13 Centerlines extracted from 2 seeds with the seedspawner tracker. All reference vessels are explored and tracked, and new uncharted branchings are found in the right arterial tree.	75
4.14 Centerlines tracking referred to the reference vessels' centerlines and radii, 2 seeds with the seedspawner tracker. This graph shows that the centerlines are correctly tracked.	75
4.15 Seeds planted by the seedspawner tracker (green) starting from just 2 seeds (yellow).	76
A.1 Cross section of the human heart. Source: Encyclopædia Britannica, Inc. [22]	97
A.2 Coronary arterial system. Source: Encyclopædia Britannica, Inc. [22] . . .	99
A.3 Heart vascular system (Netter Basic Science). Source: Frank H. Netter, Atlas of Human Anatomy, Elsevier (2018), Page 321	101
A.4 Diagram of an atherosclerotic plaque. Source: Encyclopædia Britannica, Inc.	102
A.5 Prevalence of coronary heart disease, by age and sex, United States, 2015–2018 (NHANES, National Health and Nutrition Examination Survey). Source: AHA [89].	104
A.6 Prevalence of myocardial infarction, by age and sex, United States, 2015–2018 (NHANES). Source: AHA [89].	104
A.7 Left: Annual number of US adults per 1000 having diagnosed heart attack or fatal coronary heart disease, by age and sex, 2005–2014 (ARIC Surveillance, and CHS). Right: Incidence of MI, by age, sex, and race, United States, 2005–2014 (ARIC Surveillance). Source: AHA [89].	104
A.8 Incidence of heart attack or fatal coronary heart disease, by age, sex, and race, United States, 2005–2014 (ARIC Surveillance). Source: AHA [89]. . .	104
A.9 Balloon angioplasty and stent insertion. A. In a coronary artery where blood flow is obstructed by the growth of atherosclerotic plaque, the point of obstruction is reached by a cardiac catheter encased in an inflatable balloon and wire-mesh stent. B. The balloon is inflated, thus expanding the stent, dilating the artery, and compressing the plaque. C. The balloon is deflated and withdrawn with the catheter, leaving the stent expanded against the arterial wall. Source: Encyclopædia Britannica, Inc.	105

- E.1 An example of the data with corresponding reference standard. Top-left: axial view of data. Top-right: coronal view. Bottom-left: sagittal view. Bottom-right: a 3D rendering of the reference standard. Source: Schaap et al., 2008 [72]. 119
- E.2 A geometric illustration of the terms used in the evaluation measures. The reference standard with annotated radius is depicted in gray. The terms on top of the figure are assigned to points on the centerline found by the evaluated method. The terms below the reference standard line are assigned to points on the reference standard. Source: Schaap et al., 2008 [72]. . . . 120

List of Tables

3.1	Structure of Wolternik's CNN [99]. For each layer (Layer), the convolution kernel width (Kernel width), the dilation level (Dilation), the number of output channels (Channels), and the receptive field at that layer (Field width, voxels) are listed. The number of the final output channels is equal to the number of potential directions in D, plus one channel for radius estimation. Total number of trainable parameters: $\sim 176'500$ with $ D =500$.	38
3.2	Structure of the midsplit-4 network. For each layer (Layer), the hyperparameters of the convolution filters are listed. After layer 4, the network splits into two independent networks. Total number of trainable parameters: $\sim 236'300$ with $ D =500$.	41
4.1	Results of the local test on Wolternik's network [99], standard training strategy. JNO: "just near ostia". AO: "away from ostia". α_{err} : angular error of the estimated directions. $d(P_l, P_c)$: distance between the landing point and the closest centerline point. $d(P_l, P_{c,opt})$: distance between the landing point and the point on the centerline the network was trained to land on.	68
4.2	Table showing the most relevant metrics for evaluating the tested CNNs. The meaning of the symbols are explained in section 4.1 and in table 4.1. Of all the metrics, the most relevant ones to understand how well the network is trained are $\bar{R}_{err,abs}$, $\bar{\alpha}_{D,err}$ and $ \vec{P}_{landing} - \vec{P}_{training} $, while the most relevant to understand how well the network should perform during the actual tracking task is $ \vec{P}_{landing} - \vec{P}_{centerline} $: the lowest this metric, the better the network should be able to track the centerline, on average. These and other metrics obtained from the tests on the networks (called "local tests" in this document) are discussed in details in sections 4.1. . .	71
4.3	Results of the local test on the proposed midsplit-4-FD network, standard training strategy. These results can be directly compared to the ones reported in table 4.1. Metrics are enhanced both globally and in AO conditions.	72

4.4 Table reporting the computed metrics for each test, for each vessel. The blue metrics give an idea of the tracking ability, while the orange metric AI of the tracking accuracy.	76
--	----