Automatic Deformable Modelling for 3D Medical Image Segmentation

Robert Ieuan Palmer

Submitted to Swansea University in fulfilment of the requirements for the Degree of Doctor of Philosophy

Prifysgol Abertawe
Swansea University

Department of Computer Science
Swansea University
UK
2016

Declaration

This work has not been previously accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed .... (candidate)

Date ....

Statement 1

This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

Signed .... (candidate)

Date ....

Statement 2

I hereby give my consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed .... (candidate)

Date ....

Abstract

Segmentation is the foundation of medical image analysis. Valvular segmentation is used for diagnosis and valve replacement planning, while vascular segmentation is used for diagnosing and modelling pathological function. Deformable modelling combines image and shape smoothness forces, and often uses statistical constraints for shape preservation. However for pathological organ segmentation they can be too restrictive. As such an image-driven framework which emphasises the use of image data, is proposed for valvular and vascular segmentation. To ensure an image-driven deformable model, both automatic pose estimation and boundary detection employ learning-based classifiers using image intensity data. The model's surface is smoothed with B-spline-based 3D regularisation, which is controlled by the original and new vertices after boundary detection, again meaning that the process is image-dependent. For complex valve shapes, prior-based constraints are applied for some shape preservation, however these constraints are applied non-iteratively. This ensures that the framework does not overly rely on shape priors, allowing for flexible deformation.

Although efficient at the testing stage, preparing such a system is work intensive and time consuming. This thesis also presents efficient model preparation by employing neural network (NN) classification, and a new method for automatically identifying mesh surface correspondences. The manual work of identifying suitable boundary features for both valvular and vascular segmentation, is significantly reduced by allowing NNs to *learn* such features instead of hand-crafting them. Such boundary detection produced good segmentation for both the aortic root and lymphatic vessels. Similarly, a novel combination of marginal space learning (MSL) and NNs, called NN-MSL, is capable of automatically estimating the aortic root's pose by learning sufficiently abstract features. Novel multi-resolution pooling is also employed in NN-MSL, which dramatically reduces the network's architecture to reduce training time, but still incorporates global and local structure information. In addition, statistical shape model (SSM) generation can be work intensive as surface correspondences are required across a set of training shapes. This is often done manually, or with slow automated methods that only produce sparse vertex correspondence. Here, a fast method is proposed for finding dense correspondences across a set of aortic root meshes. This is done through non-rigid registration estimated with a mesh-based similarity metric, which is shown to be more suitable than image-based registration on noisy cardiac CT data.

Qualitative and quantitative results are presented for dense correspondence identification, and the 3D segmentation of aortic roots in CT and lymphatic vessels in confocal microscopy images. They show that the proposed methods are superior to other state-of-the-art approaches for such data, and model preparation is dramatically reduced.

Acknowledgements

I would like to express my special appreciation to my supervisor Dr. Xianghua Xie. He has helped me with every aspect of my PhD studies, and has constantly nudged me in the right direction. His door has always been open for me to ask his opinion, and has frequently provided me with inspiration and guidance to carry out my research. I am lucky to have had him as my supervisor.

I am also grateful to Dr. Gary Tam for his help and support. He has frequently helped me to take a step back and think of different points of view on numerous occasions, which has proved both helpful and necessary.

I am thankful to my colleagues in the Computer Vision group for their tips and advice over the past few years. They are Dr. Huaizhong Zhang, Dr. Feng Zhao, Dr. Ehab Essa, Dr. Jonathan Jones, Mike Edwards and David George. I would also like to especially thank Jingjing Deng who has helped me with numerous tasks throughout my time in the group, and has constantly been there to bounce ideas off.

Finally I want to thank my parents for their help and patience throughout my studies. Without their support this thesis would not be possible, and I am very grateful.

I would like to dedicate this thesis to my grandmother, who passed away during my PhD study. She never failed to make me laugh. Thank you Mam.

Contents

List of Publications

The following is a list of published papers as a result of the work in this thesis.

1. Robert Ieuan Palmer, Xianghua Xie, Gary Tam, “Automatic Aortic Root Segmentation with Shape Constraints and Mesh Regularisation”, In British Machine Vision Conference (BMVC), 2015. (Oral acceptance, 7%).

2. Robert Ieuan Palmer, Xianghua Xie, Gary Tam, “Finding Complete 3D Vertex Correspondence for Statistical Shape Modeling”, In IEEE Engineering in Medicine and Biology Society (EMBC), 2015.

3. Robert Ieuan Palmer, Xianghua Xie, Gary Tam, Rob Alcock, Carl Roobottom, “Computing 3D Mesh Correspondence for Aortic Root Shape Modelling”, In Medical Image Understanding and Analysis (MIUA), 2015.

4. Robert Ieuan Palmer & Xianghua Xie, “Neural Network Boundary Detection for 3D Vessel Segmentation”, In Advanced Concepts for Intelligent Vision Systems (ACIVS), 2016.

List of Tables

List of Figures

Chapter 1

Introduction

Contents

Medical imaging is an essential branch of medicine for patient diagnosis and operative assistance. Since the first X-ray scan in 1895 [213], numerous imaging modalities have been developed for visualising organs, which includes CT, MRI, PET, SPECT, ultrasonography and microscopy [79]. The analysis of such images has traditionally been carried out manually by medical experts, which is subjective and time-consuming. Digital image analysis provides semi-automatic or automatic methods for organ visualisation, classification, comparison, measurement and function modelling. Medical image segmentation is the process of delineating an organ in 2D or 3D, which highlights its structure by essentially removing unwanted objects from the image. It is therefore used for many of the aforementioned analysis applications [238].

Parametric deformable modelling is an effective segmentation technique which offers robust segmentation for noisy images, as its deformation is constrained to ensure smooth results [275]. They are therefore composed of image-driven and shape-driven forces, which can be interchanged and their importance adjusted. In this thesis the development of automatic 3D deformable models for challenging medical data is investigated. An image-driven framework, which emphasises the use of image data rather than shape priors, is presented and is primarily applied to challenging cardiac CT images for the segmentation of the aortic root. In addition it is also applied to vascular confocal microscopy images to show the framework's flexibility.

1.1 Motivation

1.1.1 Valvular Segmentation

Valves are anatomical structures that control the flow of fluid in the body, and are most synonymously associated with the four chambers of the heart. Their performance in terms of opening, closing and the amount of blood regurgitation often defines pathological diseases, therefore accurate analysis is necessary. As a result, segmenting heart valves such as the tricuspid, mitral, pulmonary and aortic valves are necessary for cardiac diagnosis and operative planning.

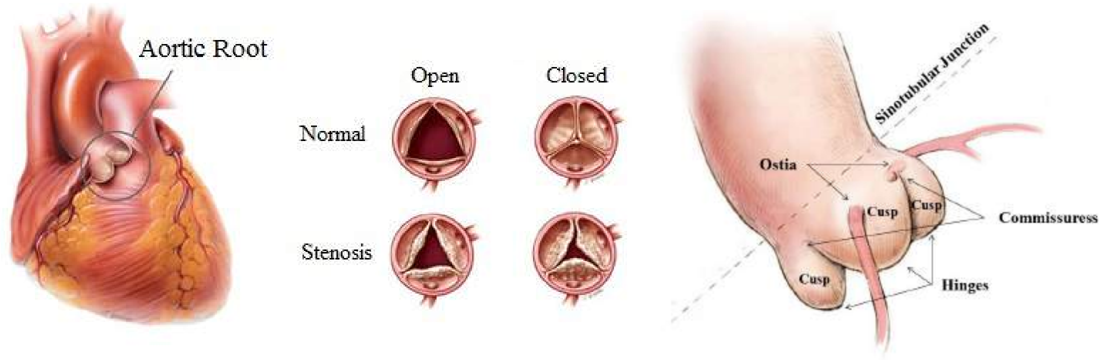


Figure 1.1: Diagrams (adapted from [105] and [47]) showing the aortic root, the difference between normal valves and valves with stenosis, and an illustration of the aortic root with structure labels.

The aortic root, which includes the aortic valve, is the meeting point between the body's main artery and left ventricle of the heart. Oxygen-rich blood is pumped from the left ventricle, through the aortic valve, the remaining aorta and onto the rest of the body. Aortic stenosis, shown in Figure 1.1, is a common heart disease where the aortic valve does not fully open, and is usually a result of calcium deposits in the artery which narrows the valve. This results in decreased blood flow from the heart, which in turn, can lead to severe hypertension and angina. Left untreated, severe aortic stenosis has a poor prognosis, leading to functional deterioration, heart failure, and often death.

Surgical aortic valve implantation (SAVI), or replacement (SAVR), is currently the standard treatment for aortic stenosis however it is estimated that up to 30% of patients suffering from the disease cannot undergo surgery, and are deemed "inoperable" [91]. However, an alternative method for valve implantation without the need for open-heart surgery has been proposed [48]. In the last 10 years a trans-arterial approach called Transcatheter Aortic Valve Implantation (TAVI) (or replacement (TAVR)) has emerged, and has been proven to be reproducible and safe [91]. First developed in 2002 by Cribier et al. [60] TAVI is now the new standard of care for patients with aortic stenosis who cannot undergo open heart surgery. The popularity of the procedure is rapidly growing, with over 50,000

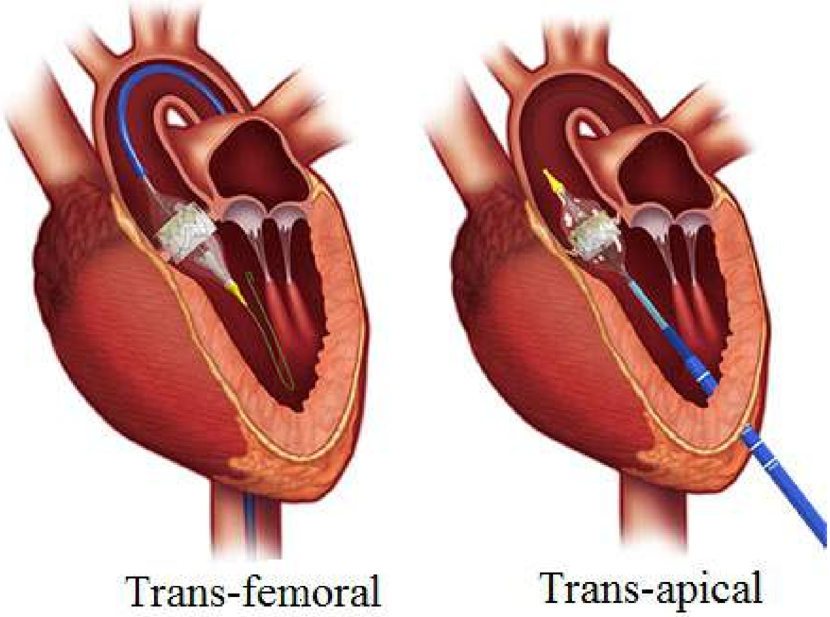


Figure 1.2: Schematic diagrams (adapted from [104]) showing both the trans-femoral and trans-apical TAVI approaches.

procedures in 40 countries to date [91].

There are two approaches for implanting a valve in the aortic root; the *trans-femoral* and *trans-apical* approaches. Both involve inserting a bioprosthetic valve through a catheter, which is implanted within the diseased valve. The trans-femoral approach involves performing a standard aortic balloon valvuloplasty, which is followed by the insertion of a sheath through the femoral artery. The bioprosthetic valve is then folded onto a balloon catheter and is transported to the native aortic valve. The balloon is inflated, allowing the bioprosthetic valve along with the support framework to simultaneously expand, and the support framework is secured to the aortic annulus and leaflets. Once the bioprosthetic valve is secured, the catheter, along with the balloon can be removed [148].

The trans-apical approach involves opening the pericardium over the apex of the left ventricle. The left ventricular apex is identified and punctured by an arterial needle, allowing the placement of a sheath through the apex into the left ventricular cavity using a standard over-the-wire technique. A wire is then advanced through the aortic valve, and a valvuloplasty balloon is then passed

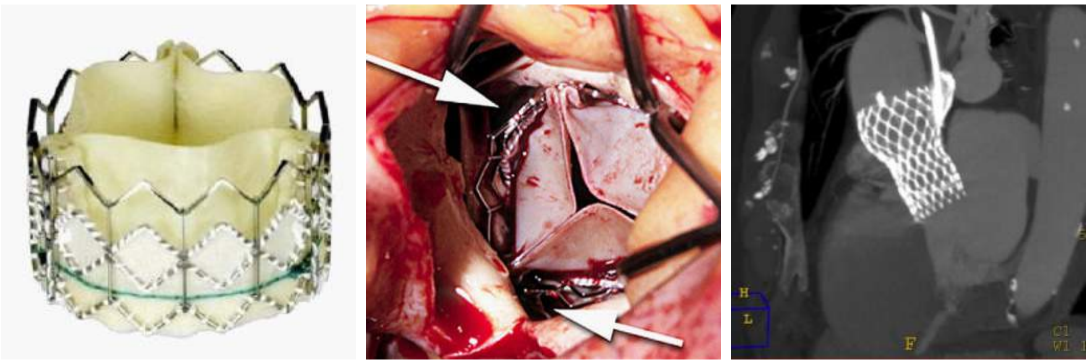


Figure 1.3: Left: Prosthetic valve (adapted from [106]). Middle: Prosthetic valve being inserted to the correct position (adapted from [106]). Right: CT image of the newly positioned prosthetic valve (adapted from [41]).

over the wire to the level of the ascending aorta where the air is evacuated before being withdrawn to the level of the valve. The catheter and bioprosthetic valve are passed over the wire and introduced through the sheath. The valve is then put into position, before the balloon, catheter and wire can be removed [158]. Schematic diagrams of both approaches are shown in Figure 1.2.

In a recent report comparing the success of the two approaches [86], early mortality appears to be higher with the trans-apical approach, and in some studies it has been shown that it can be almost twice that seen in the trans-femoral approach. A similar trend has been reported in the incidence of post-procedural stroke, and renal failure requiring dialysis, with the trans-femoral approach having a slightly lower percentage. For these reasons, the trans-femoral TAVI approach is generally the first choice for the majority of cardiologists.

Imaging is used to first of all deem whether the patient is suitable to undergo the procedure. If a patient is deemed suitable, additional imaging is used for pre-operative planning, which consists of deciding which approach to take, and the size and positioning of the valve itself. Figure 1.3 shows this. Recent developments have also used imaging modalities to intra-operatively help cardiologists and surgeons to guide the transcatheter and valve to the correct position according to the pre-operative plan [93].

As the trans-femoral approach is preferred by cardiologists, its feasibility is first analysed, followed by the feasibility of the trans-apical approach. In both cases, the path that the catheter takes must be assessed. Although the majority of the trans-femoral and the trans-apical paths are different, both paths lead to the same place; the aortic root and the ascending aorta. As a result, accurate dimensions and geometries of both are paramount. A comprehensive understanding of the geometry of the aortic root is also needed to ensure correct valve sizing and positioning. The four goals here are to; 1) determine the size of the aortic annulus; 2) determine the length of the aortic leaflets; 3) locate the coronary ostia; 4) identify other features that may interfere with the procedure.

It is well established that the aortic root is an oval shaped, 3-pronged crown-like structure, and can be said to have three hinge points, three commissure points, and two ostia. The cusps extend from the sinotubular junction down to the left ventricle itself. An illustration of these structures is shown in Figure 1.1. Given the different structures belonging to the aortic root, there are various ways of defining its diameter [17, 93]. In all cases however, accurate measurements of the root are important for the success of the TAVI procedure. The aim is therefore to develop an accurate and fast segmentation method for the aortic root in 3D cardiac CT images.

The region in the left ventricle which blood must pass to get to the aortic valve is called the left ventricular output tract (LVOT), which can be a noisy area in medical imaging. Often the aortic root and the LVOT both contain blood flow, which on a CT image has constant pixel intensity. In some cases this makes the boundary between the aortic valve and the LVOT very difficult to identify, even for manual segmentation. These challenges are highlighted in Figure 1.4. Furthermore, many patients have calcification deposits in this area, obscuring the boundary further still.

Deformable modelling is a segmentation technique which drives an initial model towards the object boundary with image-driven components and shape regularisation in order to keep the segmentation as smooth as possible. Non-deformable model based segmentation approaches are reported to be susceptible to fail in the

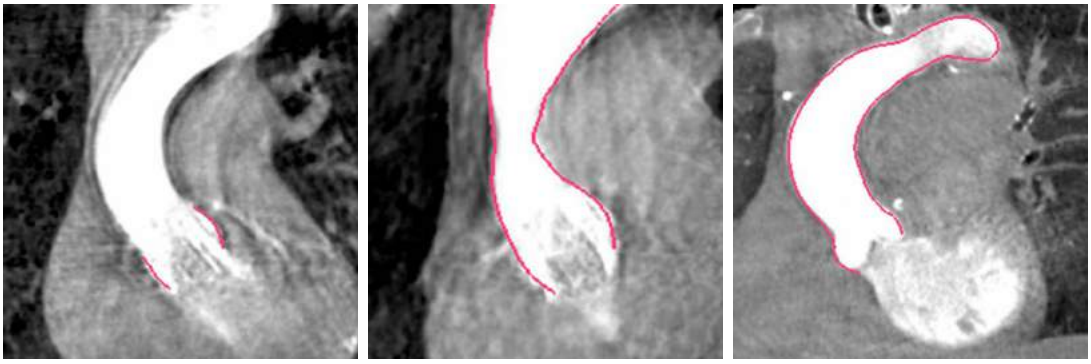


Figure 1.4: Example segmentations of the aortic root which highlights the challenges associated with cardiac CT data (adapted from [288]).

LVOT region as they can suffer from noise and intensity inconsistencies [285]. Due to their intrinsic smoothing capabilities, if an initial valve-like model has a boundary near the ground truth boundary, it is more capable of deforming towards this region, even in very noisy images [100, 285, 288]. For reasons such as this it is claimed that deformable modelling is highly suited to medical image segmentation [13, 173, 174, 176, 239, 240, 256, 275], and is assumed to be the best option for the segmentation of the aortic root.

1.1.2 Vascular Segmentation

Vessels are tubular structures which act as canals for transferring fluids around the body. Blood vessels for example include veins, arteries and capillaries, which have specific roles for healthy cardiac function [99]. Veins are thick structures that carry de-oxygenated blood from body tissue to the heart, while arteries carry oxygen-rich blood from the heart, and distributes it around the body where needed. Capillaries are the smallest blood vessels and allow both oxygenated and de-oxygenated blood to reach narrow spaces. Analysis of such structures has many uses for patient diagnosis. Accurate segmentation can help with the identification of vessel blockages and aneurysms, which can avoid significant depleted blood flow. In addition their segmentation can help identify vessel wall thinning which may avoid vascular ruptures. In terms of the TAVI procedure, vascular segmentation of the remaining arterial aorta is important for analysing the route which the catheter

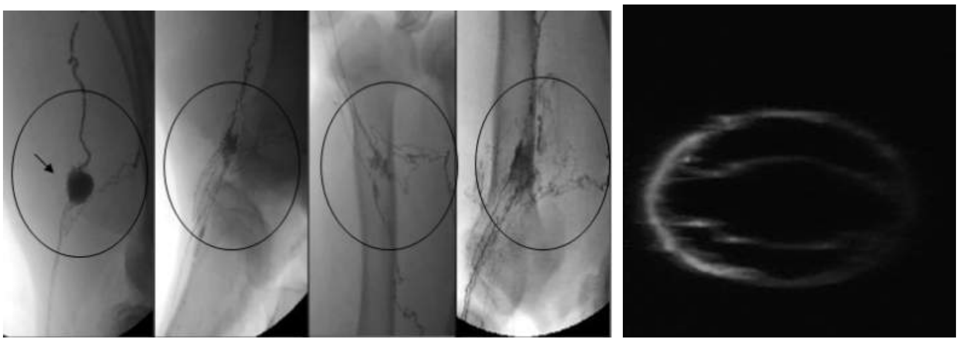


Figure 1.5: Left: Fluoroscopy images of the lymphatic vessel with entanglement and aneurysms (adapted from [10]). Right: Confocal microscopy image of the lymphatic vessel showing contrast variation.

takes to deposit the prosthetic valve. This is especially true for the trans-femoral TAVI approach, where the catheter travels through tubular structures of the descending aorta, aortic arch and finally the ascending aorta before reaching its destination.

Other types of vessels are lymphatic vessels, which collect lost fluid from capillary beds as a result of the nutrient exchange process, and transports it back to venous blood vessels. This fluid is mainly colourless and can be found in most areas of the body apart from the brain. The lymphatic vessels are mainly porous which allows the lymph fluid to enter the vascular system [99].

Lymph vessel diseases include lymphedema, which is an obstruction of lymph flow in the lymphatic vascular system. Such an obstruction can be caused by vascular aneurysms or blockage due to abnormally large fatty structures, and can lead to the lymph fluid flowing in the wrong direction back to the capillaries. With time, this can lead to significant tissue swelling and discolouring, and can cause significant pain. Analysis of early lymphedema symptoms is desired, and automatic segmentation of lymphatic vessels can immeasurably help with this diagnosis. However, lymphatic vessel walls are extremely thin and porous, making them one of the most challenging vessel types to image. Many modalities have been used for their visualisation, including infra-red fluorescence [205], and even microscopy [12, 258]. Examples of such images are shown in Figure 1.5.

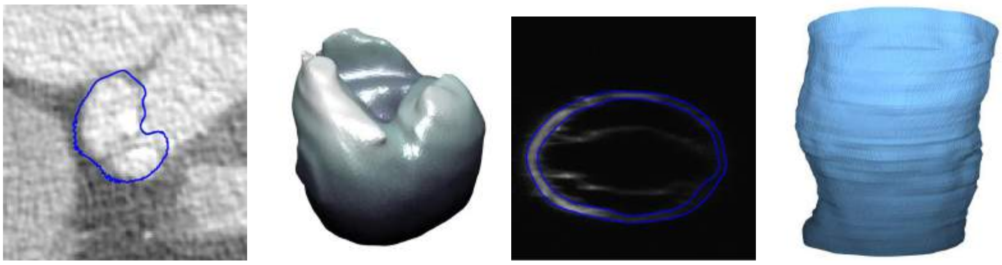


Figure 1.6: Objective outcomes. Left columns: Aortic root segmentation. Right columns: Lymphatic vessel segmentation.

Vessels may have noisy image appearance due to a number of factors. Firstly, pathological aneurysms or vessel entanglement can obscure the image significantly depending on what imaging modality is used. Other solid-like deposits such as calcifications and fats [99] and imaging contrasts can also obscure vessel walls, as well as the thickness of the wall itself. Given the nature of their potentially noisy images it is thought that the flexibility of deformable modelling is also suitable for such segmentation.

1.1.3 Objective

To this end, the main objective of this work is to develop an efficient, fully automatic segmentation method capable of accurately delineating the aortic root and vascular structures in noisy images. Examples of the objective outcomes are shown un Figure 1.6. A framework consisting of an image-based parametric deformable model is presented which is not only capable of segmenting complex valve-like structures but also vessels in challenging data. Image-driven learning-based boundary detectors are used to drive the model towards the boundary, and soft shape regularisation is employed with B-spline based interpolation. Emphasis is also put on efficient model training, by learning suitable features rather than hand-crafting them, and automating the building of a prior-based shape model.

1.1.4 Approach

Deformable models consist of image-driven processes based on image data, and soft smoothing constraints to deform the model towards the boundary while maintaining a degree of smoothness [13, 173, 174, 238]. The intrinsic shape smoothing that deformable models possess is often combined with prior knowledge for strong shape constraint [52,53,54,55,56,57,58,64,176,275]. These constraints are derived from statistical models, and force the deformed shape to resemble, with some variance, the example shapes that have already been seen in a training set. However, medical images are often taken only if the patient has pathological symptoms, therefore the organ of interest often deviates in shape to what a normal organ should look like. Furthermore, pathological organs can look significantly different to one another, depending on the disease type and severity of that disease. It is therefore assumed that in some cases, prior-based shape constraints can force the deformed model to resemble an equivalent pathological organ that looks significantly different to what it actually looks like. In other words, the strong shape constraints may restrict the model from deforming towards the true boundary. As such it is important not to overly rely on these high-level forces, and it is proposed that image-driven deformable modelling should be used for segmenting pathological anatomical structures.

Bottom-up segmentation approaches are image-driven, which means reliance on low-level image intensities such as features. In a deformable modelling context this means that the image-driven force outweighs the shape preservation force in terms of importance. The presented deformable modelling framework in this thesis only employ soft shape-constraints. This means that only soft surface interpolation is employed. In addition, in cases where prior-based shape knowledge is used, their constraints are only applied non-iteratively, allowing the image-driven forces to do most of the work.

Furthermore, model preparation and training can be very time consuming for deformable model segmentation. For example it is necessary to deduce the correct approach to drive the model, which involves choosing the correct features to identify the correct boundary. However this can be a very time consuming process, and in some cases may lead to trial and error. Other aspects of model preparation may be the generation of a statistical shape model in order to introduce some shape prior. Generating such a model can also be time consuming, and requires a significant amount of manual work due to the need for manual labelling of corresponding landmarks. To this end an image-driven automatic deformable modelling framework is presented, and additional steps have been taken to reduce the manual work needed for its preparation.

1.2 Overview

1.2.1 Dense Mesh Correspondence

To incorporate prior shape knowledge into a deformable modelling pipeline, a statistical representation of the organ's shape variability is required [52, 53]. Statistical shape models (SSMs) require a set of corresponding points across multiple shapes to be able to do this, which often requires manual labelling. The first part of this thesis is concerned with developing automatic method for establishing dense mesh correspondence between multiple shapes. Parametrisation-based methods have previously been used for finding correspondence between anatomical structures in 2D [62, 63], however they are slow to implement and impractical in 3D. Quicker methods have been proposed using image data [82,215,216] by transforming all meshes to a target mesh, and propagating landmark points across. However such landmarks are sparse, and image-based transformation is unsuitable in noisy data. An alternative similarity metric is proposed for such transformation, and correspondences are found for *all* vertices in the target mesh.

1.2.2 Aortic Root Segmentation

The second part of this thesis presents an image-based deformable model for segmenting the aortic root. To automatically initialise the process an automatic estimation of the root's pose is used, which is estimated with marginal space learning (MSL) [285, 288]. The deformation framework then consists of learning-based boundary detectors to drive the model towards the boundary with image data. This is followed by shape regularisation based on B-spline non-rigid transformations, and novel non-iterative shape priors to ensure an image-driven approach. Two implementations of the framework are implemented. Firstly a hand-crafted implementation consisting of manually defined features for pose estimation and boundary detection are used, and the SSM is manually built. Secondly an automated implementation is used, where features are *learned* rather than hand-picked, and the SSM is automatically generated. Neural network-based pose estimation is used here, called NN-MSL, which uses novel multi-resolution pooling to significantly reduce the number of parameters to optimise, and speeds up the training and testing processes.

1.2.3 Lymphatic Vessel Segmentation

Finally, a similar framework is applied for segmenting vascular structures in noisy data. In this case segmentation is carried out for the lymphatic vessel in 3D confocal microscopy images. This shows the flexibility of the framework by working on images of a different modality and alternative obscurity challenges. Instead of learning-based pose estimation, simple intensity-based filtering in the polar coordinate system is applied to generate an initial 3D model of the lymphatic vessel. Learned features are then used for boundary detection, which are capable of identifying highly inconsistent vessel wall intensities with high accuracy. Once again, B-spline based interpolation is used which ensures an image-driven approach.

1.3 Contributions

The main contributions of this thesis are as follows:

1. Non-iterative shape priors. Shape constraints from statistical models are often applied iteratively for shape regularisation [52, 53]. Non-iterative application is proposed at the initial segmentation stage. On the first deformable model iteration, large deformations can occur, which is followed by applying strong shape constraints to ensure a sensible shape appearance. For the

remaining deformable model iterations, shape constraints are not applied to ensure a more image-driven approach for pathological structures, allowing suitable deformation flexibility.

2. Efficient model training. Deformable modelling with shape constraints requires shape prior preparation as well as suitable feature identification. A combination of two automated components is proposed to dramatically speed up preparation time and reduce manual work. Firstly, the generation of a statistical shape model is automated using (3). Secondly, deep learning and NN-based machine learning is used for both boundary detection and object detection (such as (4)), which eradicates the manual task of choosing appropriate features.

3. Mesh-based similarity metric for finding dense mesh correspondence. A local transformation between two meshes is used to find corresponding landmarks between the two. A mesh-based similarity metric is proposed for transformation estimation, instead of a conventional image-based metric [82, 216], which is unsuitable for noisy data. Local transformations estimated on image data may become inaccurate in noisy images such as those of the aortic root, which ultimately leads to poor mesh correspondence. Introducing a mesh-based similarity metric avoids such inaccuracies in noisy data.

4. NN-MSL with Multi-Resolution Pooling. Neural network-based marginal space learning is proposed for estimating the aortic root's position and orientation. Deep learning is used to learn abstract representations of the data, and eliminates the need for hand-crafting features. Furthermore, novel multi-resolution pooling is employed to reduce the number of input nodes, and subsequently reduces the size of the NN architecture. This speeds up both training and testing processes. The multi-resolution aspect of the pooling allows integration of information from both large-scale, global structures such as lungs, as well as more detailed local structures such as valve cusps.

1.4 Outline

The rest of this thesis is organised as follows:

Chapter 2: Background. This chapter provides an overview of medical imaging and medical image segmentation, with emphasis on aspects of automatic deformable modelling.

Chapter 3: Mesh Correspondence. This chapter presents an automated method for finding complete mesh correspondence. An alternative mesh-based similarity metric is used for local transformation compared to conventional image-based approaches. Experimental results for both methods are provided and compared.

Chapter 4: Aortic Root Segmentation. This chapter presents an automatic deformable model approach for segmenting complex aortic root structures in cardiac CT volumes. Two implementations are presented, one with hand-crafted model training, and one non-hand-crafted solution. Results for both implementations are compared against a modified active shape model and a state-of-the-art deformable model.

Chapter 5: Lymphatic Vessel Segmentation. This chapter applies a deformable modelling framework similar to that of the previous chapter and applies it to a different challenging dataset of *ex-vivo* confocal microscopy images.

Chapter 6: Conclusion. This chapter summarises and discusses the methods presented in the previous chapters, and suggests possible extensions of the work.

Chapter 2

Background

Contents

2.1 Introduction

Numerous imaging modalities are now used for diagnosis, pre-operative planning and post-operative evaluation. The appearance and quality of each image depends on modality, machine quality, patient health, pathology, and radiographer skill amongst many others. As such different imaging modalities are used for imaging different anatomical structures, and often more than one imaging modality is used for the same organ. Multi-modality imaging is also used in some cases in order to combine highlighted regions from more than one image type.

With numerous imaging modalities and the different structural natures of anatomical organs, numerous image analysis techniques are used to semi-automatically or automatically deduce useful information from these images. Such analysis includes classification, registration, reconstruction, visualisation, and flow analysis [68]. Often these techniques require an accurate and efficient segmentation process, which highlights organs of interest in noisy images.

This chapter presents an overview of such imaging modalities and their uses, followed by an overview of medical image segmentation. This includes simplistic intensity and region-based segmentation, graph-cut segmentation and an overview of both parametric and geometric deformable models. The application of such approaches will also be discussed, before concentration turns to explicit parametric deformable models, including active contours, snakes, active shape models and deformable meshes. Finally an overview of machine learning algorithms is presented which are often used in conjunction with automatic deformable models.

2.2 Medical Imaging

Since the first X-ray image was developed by Willhelm Roentgen in 1895 [213], imaging anatomical structures has been revolutionised, and its necessity is dramatically increasing with advancing medical technologies. Medical images are crucially used for diagnosis, pre-operative planning and post-operative performance evaluation. More recent uses also include guidance for ex-vivo (outside of the body) procedures, and for in-vivo (inside the body) operations, which would not be possible if not for real-time image acquisition. With such an importance on many aspects of healthcare, numerous medical imaging modalities have been developed in order to visualise multiple anatomical structures as accurately as possible. These include systems such as X-rays and computed tomography (CT), magnetic resonance imaging (MRI), positron emission tomography (PET), single photon emission computed tomography (SPECT) and ultrasound scanning.

X-rays are photons which can be generated from a cathode tube and can be directed using lead panels. When passed through human tissue the X-ray intensity decreases depending on the tissue's molecular structure. Given that different anatomical tissues have different molecular structures, the amount of X-ray attenuation they cause are different. This difference can be recorded and manipulated to create an X-ray image [99]. Traditional scanners emit X-rays from one end which are passed through the region of interest in the patient's body, before being collected at the opposite end with a photographic film. The X-ray intensity follows the Beer-Lambert law, and is formulated as;

$$I(x) = I_0 \exp(-\mu x) \tag{2.1}$$

where I_0 is the original X-ray intensity, x is the tissue thickness and μ is the tissue's attenuation coefficient [79]. As a rule of thumb, denser tissues such as bone have higher attenuation coefficients than less dense structures such as the lungs.

Computed tomography (CT) is a further development of simplistic X-ray scanners, which takes multiple images from numerous angles in the axial plane. This increases image accuracy and has more capability to distinguish between softer tissues. Using Equation 2.1, the multiple signals from different angles can be expressed as line integrals, which can be reconstructed into an image using Fourier-based back-projection algorithms [79]. Furthermore 3D and 4D CT imaging systems have been developed. 3D CT combines multiple 2D CT images in adjacent axial planes, while 4D CT combines multiple 3D CT volumes over time. Figure 2.1 shows an illustration of a 3D CT imaging system.

In some cases, contrast agents are introduced to the region of interest in or-

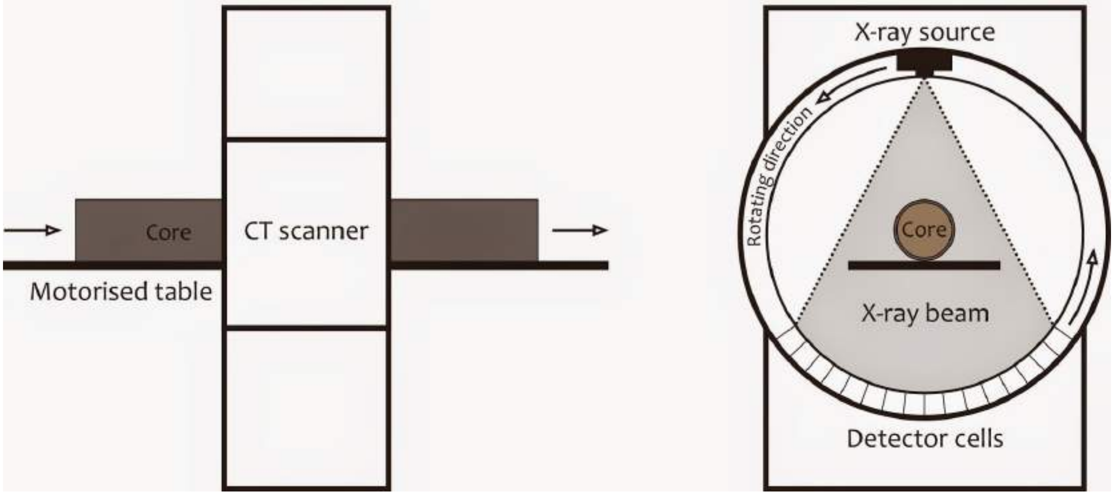


Figure 2.1: Illustration of 3D CT scanning process, adapted from [76].

der to highlight their structures. Contrast agents are substances that attenuate X-rays more than the anatomical structure itself, and are normally used when imaging small vessel or artery-like structures, that have weak attenuation coefficients. Contrast agents are normally injected directly into the patient's arteries or veins, and it is therefore essential that they are non-toxic. Iodine-based substances make good contrast agents, as they are safe and have a high atomic mass, corresponding to a high attenuation coefficient [99].

CT is a popular modality for imaging multiple anatomical structures. It has been used for brain and head scanning [8, 217], cardiac and vessel imaging [34, 172, 269], and for imaging cancerous tumours [79]. It is also frequently used for imaging the lungs for pulmonary assessment [125], and in some cases for imaging bone marrow [208].

Electromagnetic radiation is also used in nuclear medicine, or radioisotope, imaging. Unlike CT where radiation is passed through the body from an external source, radioisotope imaging uses radiation emanating from inside the body. The patient is given an emitter which is either swallowed or injected, and consists of atoms with unstable nuclei. As it decays it emits gamma rays, which passes through the region of interest and outside of the body, where they are captured by gamma cameras [79]. This captured signal is reconstructed as an image, and its appearance depends on how the emitter has been absorbed or distributed within the body [99]. Types of radioisotope imaging include positron emission tomography (PET), single photon emission computed tomography (SPECT), and 2D scintigraphy imaging.

Positron emission tomography (PET) is a popular radioisotope imaging technique, which detects positron emitting radionuclides. Positrons are positively charged particles of antimatter [255], and are emitted from proton-rich radionuclides. As they decay pairs of gammas can be detected with gamma cameras, which highlight areas of the body with higher levels of chemical activity. These are normally pathological areas, and so PET scans are widely used for revealing or evaluating cancers [99,102,200], heart disease [200], and even brain disorders [200,203].

Ultrasound imaging, or ultrasonography, is another form of medical imaging which utilises a form of radiation that both penetrates and interacts with the body. In this case the radiation is high frequency sound waves which are susceptible to reflection or refraction when encountering solid-like structures. Similar to CT, structural information is encoded within both the transmitted ultrasonic waves and their resultant scattering due to interaction with anatomical tissues. Just as the attenuation coefficient differentiates between tissues in CT, in ultrasonography a tissue's refractive index [255] determines the output signal [79].

Ultrasonography has many advantages over other radiation-based image modalities, making it one of the most popular and safest methods in diagnostic medicine [99]. Firstly, the speed of ultrasonic propagation, its measurement, and image reconstruction is relatively quick, allowing almost real-time imaging. Secondly, ultrasonic waves are low risk when it comes to patient health. This is in contrast to X-rays and radioisotopes which can be highly detrimental to patients after long exposures. Lastly ultrasound imaging equipment is small and simple, making the systems portable and cheap to build [79]. For these reasons ultrasound is commonly used for abdomen imaging, especially for real-time fetus scanning in pregnant women [186]. It is also used to asses soft tissues [188, 291], regions such as the eyes and neck [99], and the musculoskeletal system [263]. Furthermore ultrasound probes have also been placed on endoscopes for in-vivo imagery [99].

Other forms of non-radiation based systems are also available. Magnetic resonance imaging (MRI) is an imaging modality capable of describing complex molecule structures. Its process is dependent on the behaviour of protons in a magnetic field, and given that most biological tissue is abundant with water, analysis of hydrogen protons is ideal for MRI [99]. The behaviour of such protons follow the principles of nuclear magnetic resonance (NMR) [79], and is described using classical quantum theory of magnetism [98, 255].

Patients are placed in a strong magnetic field which aligns the protons in soft anatomical tissues. Radio wave pulses are exerted onto the protons, which deflects both the wave pulses and proton alignment within the magnetic field. The frequency and strength of the deflected pulse, and more importantly the time it takes for the protons to return to their original state produces a signal which can be interpreted as an image [79].

The exerted radio wave pulses play a significant role on the returned signal. By adjusting the pulse sequences, the different returning signals can be used to asses different tissue properties. Typically the different returning signals can be categorised into $T1$ and $T2$ weighted images. $T1$ weighted images show bright fatty structures and less intense dark fluids, whereas $T2$ images show prominent regions of fluid and intermediate fat signals.

Given its versatility and great capability of imaging soft tissues, MRI is used for imaging numerous anatomical structures [99]. Commonly these include imaging cardiac structures [19, 69, 268], and brain tissue [14, 217, 260]. Furthermore by employing extraordinary pulse sequences, MRI has also been used for imaging bones [5, 183, 222].

Alternative modalities include microscopy and confocal microscopy, which allows imaging of objects that cannot be sufficiently seen with the naked eye. Optical microscopy consists of emitting light through a series of lenses at a sample, and the light either transmits through the sample, or is reflected. This allows a magnified view of the sample at a specific depth. Confocal microscopy is a further

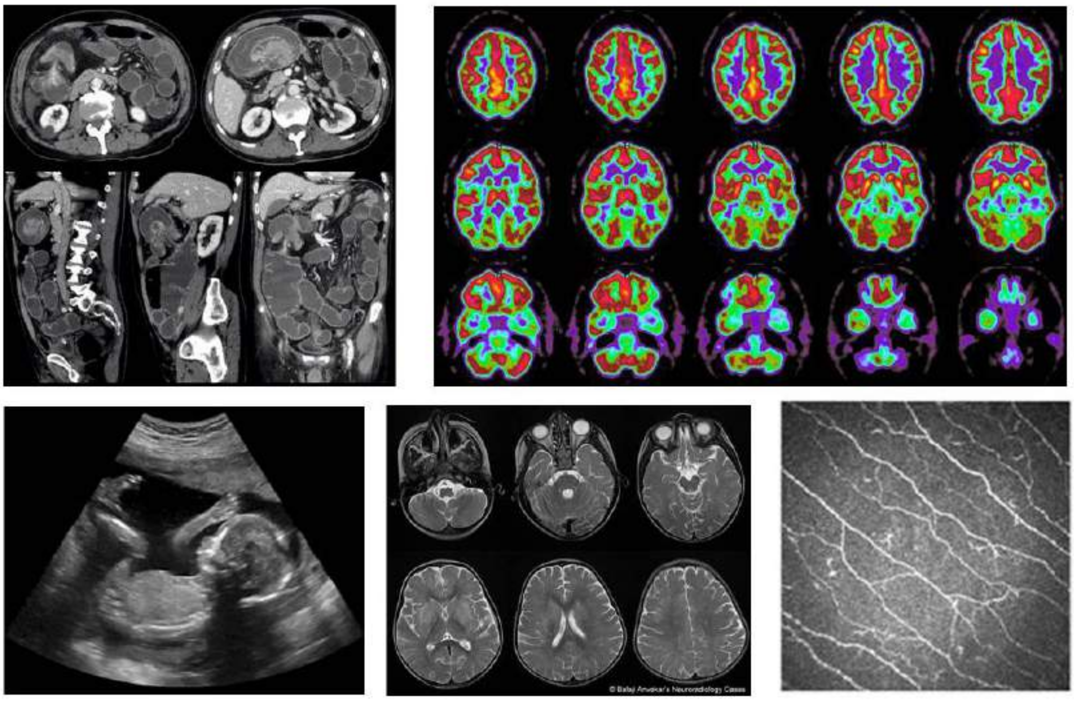


Figure 2.2: Example medical images. Top left: Abdominal CT [187]. Top right: Brain PET [190]. Bottom left: Fetal ultrasound [119]. Bottom centre: Brain MRI [11]. Bottom right: Skin confocal microscopy image [273].

development of conventional microscopy which increases the optical resolution and contrast of the image. This is achieved by adding a pinhole in front of the detector, which discards any out of focus light. As a result, sets of images at different sample depths can be taken, and can be reconstructed to create a 3D image of the sample [194]. Confocal microscopy is primarily used for cell biology, however the method has been used to image other relatively small anatomical structures, usually ex-vivo, such as eyes [40], skin [39, 204], and vascular structures [75, 226].

A summary of the advantages and disadvantages of such modalities is presented in Table 2.1.

Medical image analysis is the science involved with solving medical problems based on these imaging modalities by using digital image analysis techniques.

	CT	**PET**	**US**	**MRI**	**Microscopy**
2D	✓	✓	✓	✓	✓
3D	✓	✓	✓	✓	✓
4D	✓	✓	✓	✓	×
Invasive	×	×	×	×	✓
Ionising Radiation	✓	✓	×	×	×
Contrast Agents	✓	✓	×	✓	×
Patient Risk	✓	✓	×	✓	×
Applications (e.g)	Bones Heart Brain Cancer Vessels Lungs	Heart Brain Cancer	Bones Abdomen Muscles Eyes Kidneys	Bones Heart Brain Muscles	Vessels Eyes Skin Cells

Table 2.1: Summary of medical image modalities.

This includes classification, registration, reconstruction, visualisation, and flow analysis [68]. Image classification is used for assigning an image to a class label. In the medical context it is commonly used for determining pathological structures or determining types of diseases, which is essential for diagnosis [44]. Image registration is useful for comparing different images to highlight significant and important differences [111, 168]. This may include comparing images of the same patient from different modalities, comparing pathological structures to healthy ones, or comparing between different patients. Again, this is very useful for accurate diagnosis. Reconstruction of organs from medical images is not only used for illustration and visualisation purposes, but it is also used for organ measurement [284]. This is important for operative procedures in terms of their planning and post-operative evaluation. Furthermore, obtaining accurate measurements of in-vivo organs is again used for accurate diagnosis. Finally, medical images can be used for analysing the flow of liquids in the body, such as cerebrospinal fluid and more commonly blood flow. Flow can not only be measured but it can also be reconstructed and modelled to aid with diagnosis and operative planning [68].

The majority of the aforementioned analysis applications use image segmentation in one form or another. Segmentation is the process of dividing an image into regions that resemble objects of the real world. Medical images can contain a lot of information however in most cases only one or two anatomical structures are of interest. Segmentation highlights these structures and removes unnecessary information. This involves delineating different organs of the body for closer inspection, and for obtaining quantitative measurements of in-vivo structures [236].

2.3 Medical Image Segmentation

2.3.1 Challenges

Segmentation can generally be categorised into manual, semi-automatic, and automatic methods. Conventionally when finding objects in medical images, manual segmentation methods are used. It is thought that this results in higher accuracy as the regions of interest (ROI) are delineated by experienced medical experts such as doctors or radiologists. However manual segmentation methods have a number of disadvantages, the most obvious being the amount of time it takes to manually label a series of images. A series may include hundreds if not thousands of 512 × 512 pixel frames, and can take a number of hours to segment a single object. Another disadvantage is the subjective way in which the object might be delineated. For example, one medical expert may look at an object differently to another expert, which may yield a different segmentation result. Furthermore, a medical expert may look at an object slightly differently one day to the next depending on different knowledge or other results the expert has received. Finally, the way in which the images are displayed can affect the manual segmentation result, as parts of objects might be missed due to brightness or contrast variations.

It is hoped that automatic segmentation methods can partially avoid some of these disadvantages, which includes freeing up the time used by doctors/radiologists for segmentation. It is also hoped that segmentation will no longer be subjective, as an automatic segmentation method should obtain the same result given the same input parameters. As display is no longer as important for segmentation, it is also hoped that segmentation variations due to brightness/contrast changes can also be avoided.

Developing an automatic segmentation method however has many difficulties. Firstly, it is difficult to reproduce a computer system equipped with the years of experience that a medical expert will accumulate during his/her career. Anatomical structures are also highly variable, especially amongst unhealthy patients which makes it difficult for a computer system to delineate an anatomical structure of a particular shape if it has not encountered a structure of that shape before. The quality of the images themselves, such as noise or missing boundary information also plays an important role in automatic segmentation. For example, if there is missing boundary information, a medical expert might "fill in the gaps", however this is a difficult task for a computer system. Filters have been used to reduce noise in a series of images, however this is done at the cost of losing fine details in an image, and care must be taken to get the balance right.

Semi-automatic methods have been proposed that aim to avoid these problems while also tending towards a fully automatic system. Semi-automatic methods first involves manually identifying the object location, which involves human interaction by putting a seed in the centre of an ROI, or drawing a rough contour containing the ROI. An automatic segmentation approach is then employed to complete the delineation. Although semi-automatic schemes are currently very popular, they do not solve all of the problems encountered in manual segmentation. For example, manually choosing a seed point to initialise the process can again be subjective amongst different experts. Brightness and contrast variations of the display screen can also affect the position of the seed.

Fully automatic segmentation methods can use the same techniques used in semi-automatic methods however an additional technique to select an appropriate seed point must first be introduced (i.e the anatomical structure must be localised before segmentation can begin). Automatic object localisation is therefore an integral component of fully-automatic segmentation systems. The obvious advantage of a fully automatic approach is the lack of manual work required. This is very advantageous in medical image segmentation, where often clinical staff do not have sufficient time to manually locate objects of interest for every single image they would like to segment.

2.3.2 Methods

A number of segmentation methods have been proposed, however only a few are suitable for segmenting anatomical structures from medical images due to their high complexity. Low-level intensity-based segmentation is the simplest and oldest approach, which uses parameters such as intensity, gradient or colour to distinguish between different regions in the test image [13, 193, 212, 238]. The simple idea behind thresholding is that for an image f an intensity threshold T is defined so that pixels greater or equal to the threshold value is assigned to a specific class. Applying this over the entire image is known as global thresholding, whereas local thresholding involves splitting the image into a number of sub-images. Region-based segmentation involves using algorithms such as region-growing, split and merge, and watershed to look for groups of pixels with similar intensities [13,238]. Region growing considers each pixel as a region of its own, and are merged together if neighbours have similar intensity values [199]. The split and merge algorithm is essentially the opposite, where the entire image in considered a single region, and is split if a homogeneity criterion is not met [45]. Finally, the watershed algorithm can be thought of as a landscape which is flooded by water, and the height of the topographic plane corresponds to the intensities of the region in the image. Segmented regions are then correspondent to catchment basin boundaries [230].

Alternative methods such as classification-based segmentation have also been proposed. Groups of training pixels are labelled as either positive (boundary, foreground etc.) or negative (non-boundary, background etc.), which is then used to train a classifier [77]. Numerous machine learning algorithms are capable of such training [139], and some are described in more detail in Section 2.4. Useful descriptive features can be extracted from image regions [207] and the classifier can then classify a region in a test image as being in a certain class. Such approaches rely heavily on the image data, however classification-based systems can

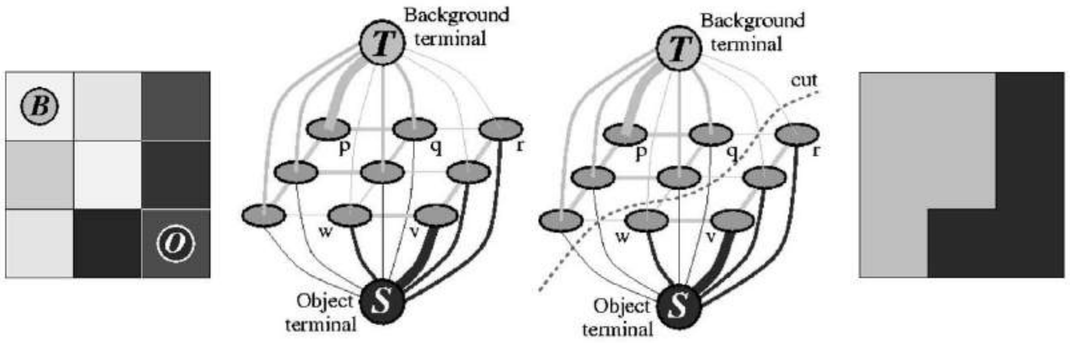


Figure 2.3: Illustration of graph cut segmentation, adapted from [26]. From left to right: 1) Original test image with seeds. B is the background and O is the object. 2) Constructed graph. 3) Graph cut. 4) Segmented image.

alternatively be used in conjunction with other techniques such as deformable modelling [173, 238].

Other approaches utilising graph-cuts have been successfully used for medical image segmentation in a number of applications [24, 25, 238], such as brain and cardiac MRI [198], liver CT [153] and vessel segmentation [75]. First proposed by Grieg et al. [101], it was suggested that minimum-cut/maximum-flow algorithms can be used for binary image reconstruction, and the idea was further developed by Boykov and Kolmogorov [27] who proposed a general framework for regional segmentation.

The method must be initiated by either manually labelling or automatically identifying one or more points in the object and background. These points serve as hard constraints, and are considered as either source s, or sink t (i.e. foreground or background). An arc-weighted graph is constructed such that;

$$G_{st} = (V \cup \{s, t\}, E) \tag{2.2}$$

where V are graph nodes which correspond to the pixels in an image, including additional terminal nodes corresponding to the initially labelled points. E are the arcs connecting adjacent nodes, and are classified as either *n-links* or *t-links*.

n-links are derived from a smoothness cost C_{smooth}, and *t-links* are derived from a data cost C_{data}. Cutting specific arcs between source s and sink t corresponds to a segmentation f. This is known as an s-t cut, and splits the nodes V into two subsets S or T (foreground or background). The solution of the graph cut is globally optimal with respect to a cost function, such as;

$$C(f) = C_{data}(f) + C_{smooth}(f) \tag{2.3}$$

where $C(f)$ is the total cost of the cut, $C_{data}(f)$ is the cost of the cut derived from the image data itself, and $C_{smooth}(f)$ is the cost of the cut derived from a smoothness term [24,25,27,238]. Figure 2.3 shows an example of graph cut on a synthetic image.

Other popular segmentation approaches are deformable models, which are curves, contours or surfaces that deform as a result of internal and external forces. Internal forces are defined within the curve itself, and define the shape of the curve by (usually) keeping the curve as smooth as possible during deformation [13,173,174,238]. External forces are computed from the image itself, with the purpose of moving the curve towards object boundaries.

Deformable models can be categorised as either parametric or geometric models. Parametric deformable models have explicit representations in the image domain, and can therefore be illustrated as having direct interaction with the image itself. Geometric deformable models however are implicitly represented in higher dimensional space with distance transform functions [176,275].

Parametric deformable models are often referred to as *active contour models* or *snake models* [128], because the convergence resembles a snake moving across the image surface. Traditionally, they can be described by an energy minimisation formulation [173,174,240,275], such as;

$$E(V) = E_{int}(V) + E_{ext}(V) \tag{2.4}$$

where V is the contour coordinates in the image domain (i.e. (x,y)), and $E(V)$ is the total energy of the model. $E_{int}(V)$ is the internal energy representing the contour's smoothness, and $E_{ext}(V)$ is the external energy representing information from the image itself. To apply such a snake to images, $E_{ext}(V)$ is attracted to image features, such as intensity or gradient extrema, that are synonymous with the boundary of the ROI. $E_{int}(V)$ can be controlled by first and second order derivatives of the curve, and therefore applies constraints to the deformation of the curve. Both $E_{int}(V)$ and $E_{ext}(V)$ can be defined in numerous ways [275], giving such models flexibility in how they are formulated.

Geometric deformable models are based on curve evolution theory, which is used to deform curves using geometric measures. Let $X(s,t)$ be a moving curve with parameters s at time t. If the curve is moving along its inward normal direction N with respect to time, it can be expressed as a partial differential equation;

$$\frac{\partial X(s,t)}{\partial t} = F(\kappa)N \tag{2.5}$$

where the speed function $F(\kappa)$ determines the speed at which curve $X(s,t)$ evolves. The curve deformation can be thought of as equivalent to the internal forces in parametric deformable modelling, which constrains the shape somewhat [275]. In a similar way to parametric models, geometric models combine the curve deformation with external image-driven forces to converge at the object boundaries.

Implicit geometric methods include the level set method [59, 170, 240, 246]. First proposed by Malladi et al. [169, 170], Sethian [229] and Caselles et al. [38], they are independent of any explicit representation. Many recent works have used extended level set approaches for medical image segmentation [145, 238, 240, 275], including CT bone scans [274], teeth [88], and for multiple organs [137] to highlight just a few. The method is associated with the curve evolution theory by defining a signed distance function, and assigning the curve at the zero level of that function, such that;

$$\phi[X(s,t)] = 0 \tag{2.6}$$

By considering the chain rule and the expression $N = -\nabla\phi/|\nabla\phi|$, its partial

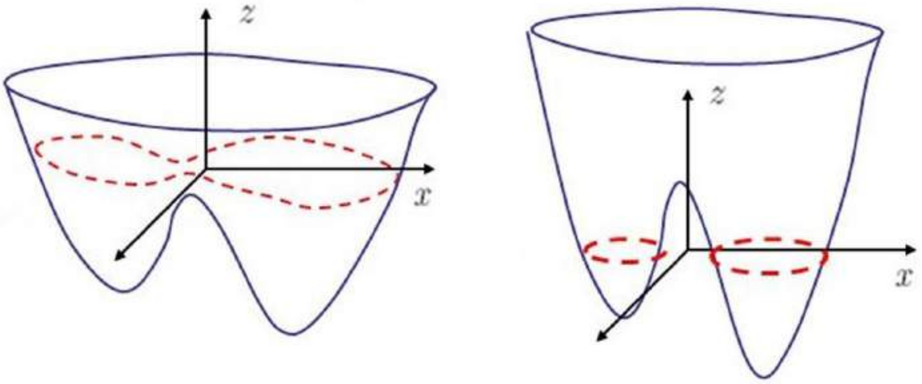


Figure 2.4: Illustration of the level set method, adapted from [49]. The level set function changes with time depending on the speed and stopping functions. The segmented contour (red dots) are always the zero level set.

derivative with respect to time can be expressed as;

$$\frac{\partial \phi}{\partial t} = F(\kappa)|\nabla \phi| \tag{2.7}$$

where $\nabla \phi$ is the gradient of the level set function. This now represents the curve evolution expressed in Equation 2.5. The speed function is a combination of constant deformations and a stopping term c, which is influenced by the image data [170]. Figure 2.4 illustrates a moving level set.

2.3.3 Discussion

Although numerous methods have been used for segmentation, not all methods are suitable specifically for delineating anatomical objects in medical images. Low-level intensity-based segmentation approaches such as thresholding and region growing are computationally inexpensive and very quick, allowing real-time segmentation. However, their performances suffer significantly from image noise. Thresholding relies heavily on the assumption that the image is bi-modal [238], which is unrealistic in most natural medical images. Region-based methods encounter similar problems, and are highly sensitive to images with high signal-to-noise ratios [77, 238]. As such simplistic intensity-based approaches are highly unsuitable for medical image segmentation.

Similarly, classification-based segmentation approaches are often deemed unsuitable for medical segmentation [238]. The primary disadvantage of classification-based segmentation is the amount of training data that is required. A large number of training samples is often required for the learned model to reach adequate generalisation. If sufficient generalisation is not achieved, the model can be prone to overfitting, yielding poor segmentation results. Often large medical image datasets are not readily available, as images are only taken for specific reasons within the clinic, and the vast majority are not made available for public research studies.

As such, more complex segmentation methods are better suited to medical image segmentation. Graph cuts, parametric and geometric deformable models have all been used effectively in this field [13,24,25,75,153,173,174,176,198,238, 239,240,256,275]. In all three approaches, shape priors can be incorporated to aid the segmentation, avoiding the difficulties that intensity and classification-based approaches face.

As a result graph cuts are relatively unaffected by intensity inconsistencies, making them suitable for segmenting potentially noisy medical images. Similarly, both parametric and geometric deformable models can be very flexible in order to cope with segmenting structures of high image and shape variability, which is often the case for pathological structures. Depending on the application, deformable models can easily be manipulated to focus more on image data or to focus more on the model's smoothness, giving them a certain suppleness when it comes to implementation [176].

However, such approaches also have unique disadvantages. Both graph cut and parametric deformable model approaches suffer from initialisation downfalls and neither can cope with topological changes, hence the development of geometric models that can. Meanwhile the geometric level-set method requires complex preparation due to its roots in curve evolution theory. The performance of graph cuts is solely based on the source and sink terminals, therefore the method is heavily dependent on the initially labelled points [150, 162]. This makes graph cuts ideally suited to interactive segmentation, where manual strokes can accurately define foreground and background pixels. Designing an automated system becomes more difficult, as these strokes must then be automatically identified [13, 238].

Parametric models heavily rely on the initial alignment of the model in the image domain [176]. Given an image of a single object, this does not have a big effect as the flexibility of the method should cope well. However in medical images a number of structures may be present, and bad alignment may result in the model converging around the wrong structure. As such they are also well suited to semi-automatic segmentation. However by incorporating a good object detection system to initialise the model, parametric models can be easily adapted to automatic systems. Like graph-cuts, another disadvantage is the need for strong shape priors, and careful design of these are needed.

Geometric methods such as level set for example, requires extensive thought into which level set functions to use, as they are often not intrinsically obvious. In addition, suitable velocity functions must also be designed for advancing the level set function [246]. The curve evolution may also continuously evolve without stopping if weak or obscured edges are present, and additional solutions to this problem must be considered.

Despite these disadvantages, parametric deformable models have many advantages for medical image segmentation, and it is this approach that is taken throughut this thesis. Despite not being able to cope with topological changes, such changes between the same anatomical structures rarely occur [99]. Therefore if the original model has the same topological landscape as the structure of interest, this should not be a problem [275]. Furthermore, parametric models are usually very quick, potentially allowing real-time segmentation [174, 176]. In most cases they also have compact representation which makes their manipulation easier, allowing more control over the result [176]. There is also scope to easily change aspects of the deformable model, such as methods for finding the boundary and methods for maintaining smooth model surfaces. This makes the options available for framework design endless. The main advantage of this is that it is possible to combine low-level, image-driven knowledge with high-level prior knowledge for balanced segmentation [240]. For these reasons, automatic parametric deformable models are discussed further in Section 2.5.

To summarise, parametric deformable modelling has been focussed on in this thesis for the following reasons.

1. Parametric deformable modelling can be very quick, potentially allowing real-time segmentation [173, 174, 176, 275]. The reason for this is that only two main components need optimisation; boundary detection and contour smoothing. Numerous approaches can be used for both, and depending on the application can be very fast indeed. This may be particularly useful for valvular and vascular segmentation, not only for fast operative planning and evaluation, but also for real-time operative guidance for in-vivo, non-invasive procedures [93].

2. Deformable modelling frameworks can be very flexible. For example individual components for boundary detection and contour smoothing can be changed depending on the application. This makes their design very flexible, and can be tailored specifically for the segmentation problem in hand.

3. Parametric deformable models have a degree of intrinsic shape preservation. For example, although they have explicit representation, they are incapable of topological changes, ensuring that if the initial model is topologically similar to the object of interest, the resulting deformed model will have the correct topology. Furthermore non-deformable modelling approaches can be susceptible to failing at noisy edges. Deformable models on the other hand may interpolate the segmentation at these areas, if surrounding model points have converged at the correct boundary. This is particularly useful for aortic root segmentation for example, where it is known that deformable models can cope with noisy regions at the LVOT [288].

4. There is no need for a topologically evolving model. Geometric and parametric deformable models have similar advantages in terms of design flexibility and intrinsic shape preservation. However, geometric models such as level sets were purposely designed to handle topological changes between the initial and resulting models. Given that it is very uncommon for pathological

valve and vessel-like structures to have different topology, there is no need to use a potentially complex geometric model for their segmentation.

2.4 Supervised Machine Learning

Machine learning algorithms allow computers the ability to learn and grow without being explicitly programmed. They do this by looking for patterns in data which adjusts the program accordingly. Machine learning is commonly used for classification problems, where given a set of input attributes a sample can be assigned a class label. This assignment is based on what patterns the program has learned from attributes it has already seen, and looks for similar patterns in new data. Such learning algorithms are often categorised as being supervised or unsupervised.

Supervised learning algorithms are given training samples, where each sample consists of a set of attributes and an associated class label. Given this information the algorithm learns what class labels should look like in terms of the attributes given to it. This is called the training process, and the classifier learns attribute patterns specifically for each class label, as well as the differences between them. After the training process a new set of attributes with an unknown class label can be passed through the classifier, which assigns the set to a class label. Numerous supervised algorithms have been used in a wide variety of applications. Examples include linear discriminant analysis (LDA), support vector machines (SVM), decision trees, classification and regression trees (CART), random forests, adaptive boosting (AdaBoost), neural networks (NN) and convolutional neural networks (CNN) [140].

Unsupervised learning algorithms has no such training process, and the algorithm is simply given sets of attributes with unknown class labels. There is no desired output in this case, and so unsupervised learning algorithms attempts to group samples with similar attributes together, without assigning them to a specific class label. This is often referred to as *cluster analysis* or *clustering* [4], and example algorithms include k-means and expectation maximisation (EM).

The obvious advantage of unsupervised learning is that known class labels are not required, however the solution relies heavily on the learning algorithm itself. Supervised learning algorithms on the other hand have some human guidance by giving training samples specific class labels. Given the importance of this in medical image analysis, supervised learning classification algorithms are desirable, and only supervised algorithms are used in this work.

A decision stump is the simplest form of classification and can be thought of as a simple condition, such as;

$$h(x) = \begin{cases} 1, x \geq 0.5 \\ 0, otherwise \end{cases} \tag{2.8}$$

which assigns sample x to a class $y = (0, 1)$ depending on some threshold (in this case 0.5). Such classifiers are conventionally termed *weak classifiers*, as a small number of parameters (in the above case just one) determine the classification of the sample. However, a single weak classifier rarely solves complicated classification problems on their own. Such classifiers can be described as searching through a hypothesis space to find an optimal hypothesis for accurate classification. However, given the available hypothesis space, even the best possible solution may not be a good one. *Ensemble learning* is an approach which essentially searches multiple hypothesis spaces. The assumption is then made that combining the best hypothesis from each search space produces a better overall solution. Ensemble methods can be categorised in two; bagging algorithms, and boosting algorithms.

Bagging can be thought of as model averaging, which combines many classification models and takes their average result. This reduces the variance in the results and helps to avoid overfitting. First developed by Breiman [29], *bootstrap aggregation* produces multiple training sets by separating the training samples into subsets, and trains one model per subset. Each model assigns a class label to a testing sample, and a scoring system is used to determine the overall classification. Conventional models can significantly change a sample's classification even if the training set is only slightly perturbed. Bagging has been shown to improve accuracy by reducing this instability through aggregated classification [29].

An alternative ensemble approach is boosting. By calling the most useful weak classifiers and combining their weak rules, a strong classifier can be built. It is then assumed that its combinatorial rule is more accurate than any of the individual constituent weak rules [261]. Such systems can be very powerful approaches for classification problems. Weights of certain weak classifiers are increased or decreased depending on how well they separate the data. Their emphasis is to let weak classifiers function sequentially, and letting certain samples that are difficult to correctly classify determine the biggest weight changes to the system.

Deep learning is an alternative learning approach to conventional machine learning algorithms such as ensembles. It is a process where much higher-level, abstract patterns are found in the data by using multiple processing layers and nodes. Such networks can be very complex and may be composed of multiple non-linear transformations. These algorithms take a different approach to conventional machine learning algorithms. Conventional algorithms use algorithmic approaches, in that the system follows a set of explicit instructions in order to solve a problem. Unless these specific steps are known (i.e. which features to extract etc.), these algorithms may fail. Therefore conventional machine learning algorithms require some knowledge on how to solve the problem in the first place. Given their ability to learn important parameters, deep learning algorithms are useful for learning patterns which we don't exactly know how to do ourselves. As such they are widely used in pattern recognition for their ability to learn from unknown and unexpected pattern information [103, 238].

The following sections present overviews of some supervised machine learning algorithms. They are algorithms that are referred to in this thesis, and represent a bagging approach, a boosting approach, and a deep learning approach.

2.4.1 Random Forests

Decision trees are made of multiple decision stumps (Equation 2.8) in a hierarchical structure. If $h(x) = 1$ after asking Equation 2.8, we may then ask if $x < 0.75$? They separate the subsets further until the bottom node of a branch, called a leaf node, yields a subset of samples of the same class label. Algorithms such as Hunt's algorithm [247] grow trees by comparing the node's output with its class label. If a node yields a subset with multiple class labels, child nodes are introduced to separate the data further. This process is repeated until all leaf nodes produce subsets of the same class label. A decision tree is an ensemble of multiple decision stumps. Random Forest (RF) takes the ensemble to a higher level, where a decision forest is an ensemble of trees. Multiple trees are grown from random subsets of the training data (bagging), hence the term *random forest*. After each tree has assigned the test samples a class label, a scoring system is used for aggregation.

Furthermore, not only are the subsets of training data randomised to grow each tree, but the way in which the trees are grown also has randomised elements [29]. Conventionally, decision trees are grown by conducting exhaustive searches across all weak classifiers to find the weak classifier which best partitions the data. This is done for every node, making it a highly greedy algorithm. Randomness is introduced in RFs by randomly picking a subset of weak classifiers for each node, and the weak classifier that best partitions the data from this subset is used as the node for that tree. Breiman [29, 30] notes that this not only means that the trees are grown from random data, but they are also grown in a randomised way. This ensures that all of the trees in the forest are sufficiently dissimilar to one another, allowing a more generalised classification after aggregation.

Two parameters must therefore be chosen for random forest training. The first is the number of trees, and the other is the amount of weak classifiers allowed in the randomised subset to identify each node. Typically the number of trees to grow is between 200 and 500, which is a number used in many implementations [29, 30, 61]. If n is the total number of weak classifiers available, the number of random weak classifiers used to choose each node has been recommended by Breiman [29, 30] and Cutler [31] to be one of; $\sqrt{n}$, $n/2$, $2n$ or $\ln 2$.

Random forests are known to be accurate for most learning problems due to their ability to generalise well. In addition, although it can handle thousands of input variables, which is more than enough for most learning problems, the algorithm can still run efficiently during the testing stage. However, due to it relying on the weak classifiers available, it is susceptible to overfit on noisy classification tasks [232].

2.4.2 Adaptive Boosting

One of the most popular boosting algorithms is AdaBoost and is widely used for image classification [124, 261]. First proposed by Freund and Schapire [84, 221], adaptive boosting (AdaBoost) finds the most relevant weak classifiers (i.e the ones that produce the least amount of misclassified samples), and adjusts their weights accordingly. The best weak classifiers are linearly combined to produce a single strong classifier [84].

Conventional AdaBoost, often referred to as *Discrete* AdaBoost uses a predefined number of weak classifiers K and adjusts their weights to minimise the error between the system's output and the expected output [261]. Assume a set of T training samples with feature vectors x_i of length m, and expected output labels $y_i = (0, 1)$. Each round of boosting k selects one feature from all possible m features, until K is reached. For initialisation, weights are defined as $w_i = \frac{1}{m}$, and are normalised. Then, on the first round each feature is defined as a feature stump h_i (i.e. weak classifier), and its error with respect to w_t is computed as;

$$E_i = \sum_{t=0}^{T} w_i |h_i(x_{t,i}) - y_i| \tag{2.9}$$

The weak classifier h_i with the lowest error E_i, now referred to as h_k, is chosen as the first of the K weak classifiers, and the weights are updated;

$$w_i = w_i \Omega_i^{1-\epsilon_i} \tag{2.10}$$

where $\Omega_i = \frac{E_i}{1-E_i}$, and $\epsilon_i = 0$ if x_i is classified correctly, and $\epsilon_i = 1$ if it has been incorrectly classified. The systems output is defined as the sum of the weak classifiers;

$$h(x) = \sum_{k=1}^{K} \alpha_k h_k(x_k) \tag{2.11}$$

where $\alpha_k = \log \frac{1}{\Omega_k}$. Binary classification is finally determined based on the following condition;

$$H(x) = \begin{cases} 1, h(x) \geq \frac{1}{2}\sum_{k=1}^{K} \alpha_k \\ 0, otherwise \end{cases} \tag{2.12}$$

Since its first implementation, a number of variations have been developed. Instead of producing a binary output. *Real AdaBoost* [221] assigns a real number score, where its sign represents the class label and the magnitude represents its classification confidence. Alternatively, *Gentle AdaBoost* fits the model in a much simpler way by minimising the squared-error loss [85]. This makes AdaBoost "gentler" as there is a limited range in which the function can be updated. This avoids very fast increases in weight adjustments, avoiding very high errors [85,147].

AdaBoost is relatively simple to implement, and only one parameter is needed to tune, which is the number of weak classifiers to combine K. They are also known to provide good generalisation [238], which subsequently allows the algorithm to be used as a feature selection method. The biggest disadvantage is that they heavily rely on the weak classifiers themselves, and so correct feature extraction is paramount. If they are too complex, boosting can lead to overfitting, however if they are too simple it can lead to under-fitting and becomes too generalised [89]. As such they may suffer greatly from noisy data. In addition, training such classifiers is time and computationally expensive, and traditional boosting algorithms can also be time consuming at the testing stage. The reason for this is that *all* selected weak classifiers must be evaluated to make a classification decision. However, alternative implementations have been proposed to speed up the testing stage by converting the boosting classifier to a deeper structure, such as a decision tree [134]. By making each tree node a boosting classifier in itself, samples that are easy to classify are done so quickly by only evaluating a few weak classifiers. However, it is generally considered that boosting algorithms perform very well if the weak classifiers are very suitable for the problem. But alternative ensembles such as bagging are simpler methods, and easier to get right [238].

2.4.3 Neural Networks

Neural networks (NNs) are composed of layers where nodes in adjacent layers are linked together with weighted connections [103, 143], and increasing the number of layers in the network creates ever deeper systems. It is a learning system that is based on the interconnections of elementary neurons in the human brain [238].

Node i in layer k is connected to node j in layer $k+1$ by weight w_{ji}^{k}, and is given a bias term b_j^{k+1}. The activation of a node at layer k is computed and used as an input for the nodes in layer $k+1$, and is repeated until the output activations are computed. An activation a_j^{k+1} is computed by calculating the weighted sum of the node's inputs, and passing it through an activation function (usually a sigmoid function) f, and is formulated as

$$a_j^{k+1} = f(b_j^{k+1} + \sum_{i=1} \mu_i^k w_{ji}^k) \tag{2.13}$$

where μ_i^k is the node's input. The network fits a function to a supervised output, where w_{ji}^k and b_j^{k+1} are optimised, meaning the network is capable of learning useful features. Increasing the number of nodes in a layer increases the number that can be combined for a node in the next layer. Increasing the number of layers in the network increases the complexity of node combinations, leading to ever more abstract features. Too few layers, and nodes in layers can lead to a network that is too generalised, whereas too many can easily lead to overfitting the data. A balance between the two must be found depending on the data used and its application.

Networks are trained by passing multiple training instances of input values through the network to get a vector output, and comparing it to the expected output. Given this difference the network weights are adjusted using back-propagation [103]. Assume a set of t training samples. A single training instance i has an array of n training inputs x^i and m expected outputs ω^i. The network has been initialised by a random set of l interconnecting weights w. Passing x^i through the network using Equation 2.13 yields an output vector y^i, and its error relative to the expected output is computed by calculating the sum of squared difference

(SSD) between the two.

$$E = \sum_{i=0}^{t} \sum_{j=0}^{m} (y_j^i - \omega_j^i)^2 \tag{2.14}$$

Given the network architecture, E is calculated through a combination of node functions. This makes the error a continuous and differentiable function of all l weights $w = (w_1, w_2, \ldots, w_l)$. As a result, error E as a result of weight w_q can be expressed as the partial derivative;

$$\Delta w_q = -\alpha \frac{\partial E}{\partial w_q} \tag{2.15}$$

where α is the learning rate which controls how severely the weights are adjusted. The weights can then be simply adjusted using $w_q = w_q + \Delta w_q$. One of these processes is called an epoch, which can be looped until convergence is found. $\frac{\partial E}{\partial w_q}$ is subsequently computed by considering the error associated with the nodes in the adjacent upper level.

$$\frac{\partial E}{\partial w_q} = \delta_j a_i \tag{2.16}$$

Where a_i is the output of node i. $\delta_j = y_j(1 - y_j)(\omega_j - y_j)$ for a weight connected to an output node, and $\delta_j = a_j(1 - a_j)(\sum_l \delta_l w_{jl})$ for a weight connected to a node in a hidden layer [238]. These parameters are minimised by using an iterative optimisation algorithm. Conventionally this is either the gradient descent or the Levenberg-Marquadt algorithm [103].

Convolutional neural networks (CNNs) are extensions to NNs which force spatial connectivity on the features. The assumption with CNNs is that we are dealing with image-like classification problems. With conventional NNs the intensity of each image pixel corresponds to individual nodes with their own weight, meaning they are completely different features. However CNNs assume that if a local feature is useful at one part of the image, it must also be useful at other parts of the image, and therefore must share the same weight. This essentially means that “identical copies” of spatially invariant features are created across all images [143].

CNNs consist of *convolutional* and *pooling* layers. A convolutional layer performs the convolutional operations based on filters. These filters are iteratively adjusted in the same way that weights are adjusted in NNs. Convolutional layers create multiple feature maps, yielding a large number of parameters to be optimised. Pooling layers reduce the size of the feature maps with algorithms such as *max*, *min* or *mean* pooling. As a result, CNNs create simpler, more generalised versions of the original image. This, along with the reduced number of parameters to be optimised make them especially useful for image recognition problems [143, 144, 234].

The advantage of deep learning is its ability to *learn* suitable features. A conventional hand-crafted feature (Haar, HOG etc.) at a given region is essentially a combination of raw intensity values around that region. The importance of these intensities vary depending on what feature is being extracted. This is mimicked by deep learning networks, as a node in a hidden layer is a combination of activations from the previous layer. The importance of each node varies by iteratively adjusting their weights. This capability means that features do not need to be explicitly hand-crafted before being inputted into the network, saving computation and manual work. However deep learning algorithms are slow to train as a result of the amount of parameters to be optimised. Furthermore, there is no straightforward way of finding a network architecture that sufficiently solves the problem. This often requires trial and error which is extremely time consuming. As a result architectures are susceptible to being overly complicated, which can lead to overfitting.

2.5 Automatic Deformable Modelling

The majority of parametric deformable modelling approaches involve aligning an initial model with the test image, before deforming the model towards the object boundary. Accurate alignment is paramount for good segmentation, and often this is manually carried out, making the process semi-automatic. Automatically estimating the pose parameters (i.e. position, orientation and scale) of an object in a test image is what distinguishes between semi-automatic and automatic deformable modelling. As such automatic deformable modelling has two distinct parts; 1) pose estimation and 2) model deformation.

2.5.1 Pose Estimation

Pose estimation can be thought of as an object localisation problem, and is the task of estimating pose parameters such position, orientation, and scale of an object in the test image. As well as being used for fully automatic segmentation, automatic object localisation can also be used for video surveillance and image retrieval. It has been implemented through a number of approaches such as template matching, learning-based, transfer- learning and patch hypothesis testing. Less research has been carried out for object localisation in medical and in particular cardiac images, however two main methods have emerged; the template matching with the generalised Hough transform [116], and learning-based patch hypothesis searching.

Template matching is the process of matching parts of a test image with a template image. These two components are paramount for implementing such object detection. The first component is a test image; the image in which we would like to find the object of interest. The second component is the template; an image patch, usually smaller than the test image, with which parts of the test image are compared to. For object detection or pose estimation, the template is an example image of the object we would like to detect. A relevant example might be the test image being a 3D cardiac volume of the torso, in which we would like to find the aortic root. The template would then be a much smaller 3D image (patch) of an example aortic root. The parts of the test image with the best match to the template given some similarity distance metric, are then taken as the locations of the object in the test image, and as such the goal is to identify all input image locations at which the template is present. Many approaches have been taken to find matches between different image characteristics [112, 113, 118, 242]. Examples include native template matching, grayscale-based matching, pattern correlation matching, image correlation matching and edge-based matching [195]. In some cases, advanced template matching systems are capable of finding template matches in the test image regardless of the object's orientation and brightness, which is a strong advantage for object detection [195]. However, for pose estimation this is problematic, as we would like to know at what orientation is the best template match.

A classic example of template matching is the generalised Hough transform [116], which has conventionally been used to approximate lines and circles in a 2D image. However more recently it has also been used to approximate more arbitrary shapes, as well as 3D objects [22]. Localising these shapes involves a voting procedure in the parameter space with the help of an accumulator matrix. To approximate a circular shape, an edge detector must first be applied in order to obtain an edge map, where high map values correspond to edge points. Conventionally, the Canny edge detector [37] is used, [9, 22] however other edge detectors have also been used, such as the Sobel filter [22]. The idea then is that a perpendicular line from each of the edge points should cross the centre of a circle. Drawing lines to every edge point then obtains hot spots corresponding to the centre of the circles. Although effective, especially for round-like structures, it is known that the Hough transform is extremely slow due to it's voting system [238], and not suitable for systems aiming for real-time pose estimation.

An extension to the generalised Hough transform has recently been proposed, called Hough Forests [87], which dramatically speeds up detection. Whereas the conventional implementation provides good generalisation by combining image parts observed by multiple training templates, a large codebook is required to convert the observed features into a Hough vote. This process significantly hinders computational performance. Instead, Hough Forests combines the generalised Hough transform with Random Forests in order to learn a direct mapping between the patch and the Hough vote, skipping the cumbersome codebook altogether. The Hough forest is learned given a set of training images, and so at runtime the patch is passed through the Hough forest, yielding the resulting Hough votes and subsequently finding the maximum likelihood of the objects pose. Such systems have been successfully implemented for detecting humans, animals and cars, amongst many other objects [87, 185, 249].

While the Hough transform relies heavily on prominent object features that are highly textured, alternative template matching methods have concentrated on improving detection of objects that are texture-less [28,112,113,131,141,184,209,272].

Like the Hough transform, other traditional template-based approaches match sparse local descriptors such as SIFT features [163], which are easily identifiable for matching. Texture-less images however have no distinct features such as lines or edges, and generally have poor visual appearance. The reason for such images are often due to image acquisition conditions, such as lighting, physical occlusions or equipment inconsistencies. To counter this, modern and current approaches to deal with these problems often resort to machine learning in order to learn abstract representations of features associated with the occluded object, rather than relying on prominent and obvious feature representations [28, 177, 184]. Learning algorithms have been used in this context in multiple ways. Firstly they have been used for learning interest features and descriptors [115, 214, 270], rather than manually hand-crafting them, and they have also been used for predicting the matches [18, 23, 28, 149, 184, 191, 192].

The recent success of deep learning for training non-linear models as a result of massive amounts of data and speedy processing, has lead to it being a hot topic, and a popular method in many computer vision problems. Notably, as with the current trend, deep learning has also been used in much the same way as previous conventional machine learning algorithms for pose estimation. Convolutional Neural Networks (CNNs) have been used [122, 131, 141, 233, 272] to learn powerful descriptors which are not intuitive. Such work has demonstrated that abstract representations can learn features that are potentially more discriminative, allowing matches to be identified in a clearer, more polarised way.

CNNs have also been used to learn features, however the network does not necessarily need to be trained on a set of relevant training images [67, 95, 224, 231]. Known as *transfer learning*, a network can be pre-trained on a previous dataset, but the features learned during this training are still capable of identifying different objects in a new dataset. The motivation behind such an approach is that training CNNs (or any other deep learning models) requires large amounts of training data to ensure good generalisation. This can often be impractical, as acquiring an abundance of annotated training data is time consuming and often expensive. As such, learned features on a previous, unrelated have essentially been reused on new datasets, to avoid such impractical training requirements. The approach has shown to be effective for objects with high texture [224,231], however the assumption is that features used to identify one type of object are abstractly similar to the features used to identify another type of object. Such an assumption means that pre-trained models need to be carefully chosen, which may not be a trivial task.

Template matching approaches have also been used specifically for finding the aortic root. Kurkure et al. [142] use the generalised Hough transform to locate the aorta in a 3D CT Volume, by approximating the circular shape of the aorta in a series of 2D axial images. They report that the dynamic programming using the generalised Hough transform performed well, however encountered some difficulties due to either a lack of edge features or noise. This was particularly noticeable when the pulmonary artery produced stronger edge responses in the edge detection stage, than the aorta itself. Avila-Montes et al. [9] used an entropy-based cost function between axial slices. This constraint forced the boundary to follow the homogeneous path through the pixels. Low local entropy corresponded to a homogeneous region, however high local entropy represented regions that were not textually uniform such as boundaries. This resulted in a better transition between adjacent pixels and pixel regions, and had a positive impact in localising the aorta. As a result the method was also used to localise the aorta by a number of authors [262,264]. Waechter et al. [262], Kurkure et al. [142] and Avila Montes et al. [9] have shown that localising relatively small objects such as the ascending and descending aorta in 3D volumes is possible using the Hough transform. However in all cases it was assumed that cylindrical cross-sections of the object were visible in each image slice. This assumption cannot be made when localising the majority of anatomical structures. This assumption may also not hold for the aorta in pathological cases, and so an alternative method is desirable. Furthermore, these examples are implementation of the generalised Hough transform, which is known for it's slow voting procedure [238].

Template matching approaches are flexible, as a vast selection of suitable features or regions can be used to create the template. Simple systems are also straightforward to implement, and their complexity can dramatically increased by combining with learning algorithms, making them a popular and powerful method for object detection and pose estimation. However, given the need to find the matches in all possible combinations of location, orientation and scale in 3D pose estimation, and given the size of medical volumes in which we need to search, it is envisaged that such an approach would be time consuming. As such an approach to reduce the number of pose hypotheses is desirable.

Patch hypothesis testing are classification-based methods that can often be thought of as object detection problems, where an object is either in a window at a certain pose hypothesis, or not. Here unknown test images are assigned to one of a set of known texture classes, and is popular in a non-medical context for detecting faces [189, 223, 261, 266], vehicles [33, 154], and humans [228, 279]. Traditionally, sample test images are usually labelled either positive (object present) or negative (object not present). Scalar descriptions called feature vectors are extracted from each sample image, and a classifier is trained to find which features best discriminate between positive and negative training windows. Once trained, features from test images are passed through the classifier, and images with high detection scores correspond to images that contain the object.

Although similar, patch hypothesis testing involves splitting the image into a large set of pose hypotheses. These can be position, orientation or scale hypotheses, depending on the application and dimension of the images. Sliding window approaches are often used in these cases, which involves extracting features from a sub-window of an image. The window is then moved along the image (i.e. moved along to the next position, orientation or scale hypothesis), and the process is repeated until all possible hypotheses have been exhausted, extracting features at every stage. Similar to object detection, the problem is then formulated as whether the object is in any of the sub-windows or not. To do this, the feature vector from each patch hypothesis is tested by a trained classifier, giving it a detection score. The window with the highest detection score represents the window that contains the object [261].

Some work has been presented which adopts learning-based detection techniques for localising anatomical structures [61, 92, 114]. The primary drawback to this approach in medical imaging is the sheer number of hypotheses to test. Consider a 2D scan I of size 512×512. The number position hypotheses alone would be $\sim 260,000$. This number would exponentially increase with increasing image dimension, or the need to estimate further pose parameters such as orientation or scale. This has a dramatic effect on the speed of pose estimation during testing, especially for 3D volumes.

Further developments have been proposed to alleviate this problem in the form of marginal-space-learning (MSL) [286]. The authors propose a technique to drastically reduce the computational burden of 3D exhaustive search for estimating position, orientation and scale in large medical image volumes. The 9 pose dimensions are estimated incrementally, which leads to efficient search in a much denser computational grid that is otherwise not possible with exhaustive search. The idea of MSL is that the full similarity search space (position-orientation-scale) can be marginalised into position, position-orientation, and position-orientation-scale spaces. As such, three classifiers are built. One to estimate the most likely position estimations, one to estimate the most likely position and orientation combinations, and finally one to estimate the most likely position, orientation and scale combination. Conventionally, these classifiers have been trained with a boosting-based machine learning algorithm [257]. Assuming some natural alignment which is often characteristic of anatomical structures, it is assumed that the optimal pose hypothesis is contained within the highest probability hypotheses of all marginal spaces. As a result the number of pose hypotheses to test is significantly reduced, which speeds up the testing stage no end. The method has been effectively used for detecting many anatomical structures [132, 285, 287, 289, 290], and in particular for small cardiac structures in very large cardiac volumes of the torso [100, 288].

As is the trend with other pose estimation approaches, marginal space learning has also been combined with deep learning for powerful abstract representations of the image patches. Ghesu et al. [94] combined deep learning with MSL for object detection in ultrasound images, which showed good performance in 2D. Their method is referred to as marginal-space-deep-learning (MSDL), and raw intensity features are inputted into a large neural network (NN) for feature learning. A complex architecture is presented with a high number of hidden layers, ensuring highly abstract representations. Furthermore, an additional cascade is utilised to try and simplify the architecture. Although training was slow due to the size of the network used, testing time is fast. An exhaustive search would not be capable of detecting such structures in such large volumes anywhere near real time, and as such MSL with conventional machine learning and with deep learning, is considered further in this thesis.

2.5.2 Active Contour Models

The origin of deformable models is with active contour models, or snakes. First proposed by Kass et al. [128], Terzopoulos et al. [250, 251, 252] and Witkin et al. [271], they deform a contour in the image domain based on image data forces and curve smoothness forces. Although predominantly originally used for 2D segmentation, as early as 1988 it was also shown that they may be easily extendible to 3D [178, 253].

Active contour models are parametric contours or curves that are geometrically positioned in the image domain, and can be represented as $V(x(s), y(s))$, where $x(s)$ and $y(s)$ are the image I coordinates on which the contour lies, and $s \in [0, 1]$ is the parametric domain. The contour's shape and coordinate locations are determined by an energy function [174, 238];

$$E(V(s)) = E_{int}(V(s)) + E_{ext}(V(s)) \tag{2.17}$$

which considers a contour smoothness term $E_{int}(V(s))$ and a term representing the image data $E_{ext}(V(s))$. Minimising this function results in a global solution to the segmentation problem. The internal energy can be written as;

$$E_{int}(V(s)) = w_1(s)\left|\frac{dV}{ds}\right|^2 + w_2(s)\left|\frac{d^2V}{ds^2}\right|^2 \tag{2.18}$$

where w_1 represents the elasticity of the contour, and w_2 represents the rigidity [13, 174, 238]. The external energy term connects the snake to the image I, which can be formulated depending on the segmentation problem. For illustration, a very simple example may take the form [238];

$$E_{ext}(V(s)) = w_{line}E_{line}(V(s)) + w_{edge}E_{edge}(V(s)) \tag{2.19}$$

where

$$E_{line}(V(s)) = I(x(s), y(s)) \tag{2.20}$$

$$E_{edge}(V(s)) = -|\nabla I(x(s), y(s))|^2 \tag{2.21}$$

and w_{line} and w_{edge} are their associated weights. Such expressions would therefore attract the contour towards line and edge features in I. From calculus variations the minimisation function can be expressed as an Euler-Lagrange equation [13, 174, 238];

$$-\frac{d}{ds}\left(w_1(s)\frac{dV}{ds}\right) + \frac{d^2}{ds^2}\left(w_2(s)\frac{d^2V}{ds^2}\right) + \nabla E_{ext}(V(s)) = 0 \tag{2.22}$$

This partial differentiation equation clearly shows the balance between internal and external energies that occurs when the contour is at an equilibrium. Numerical algorithms [174] are then often used to optimise this formulation.

The computation of the external force is highly dependent on the problem in hand, and numerous computations have been proposed [275]. Traditionally, a Gaussian potential force was used, which is designed to lead the contour towards image edges or lines, and has been expressed in the following form [251, 275].

$$E_{ext}(V(x, y) = w[G_\sigma(x, y) \star I(x, y)] \tag{2.23}$$

where G_σ is a 2D Gaussian function with standard deviation σ, and w is a weighting parameter. The weight's sign corresponds to either dark or light edges in image I. However such a function requires that the initial contour be initialised very close the ground truth object in the image domain. As a result the computation of the external force has been further developed to attract the contour to edges over longer distances. Examples include multi-scale Gaussian potential [128, 252], pressure force [50], distance potential force [51], gradient vector flow [276, 277], and dynamic distance forces [65, 167].

Although popular, the minimisation of the contour's energy can be problematic. Numerous parameters such as weights, the number of iterations, and the external energy function E_{ext} must be carefully designed for optimal performance [174, 240]. Furthermore, the Euler-Lagrange equation suffers from numerical instability [238], and so while energy minimisation is still a well used method today, other deformable models have also been proposed without such formulation. Instead image-driven forces and shape forces are applied separately and sequentially [100, 285, 288] (see Section 2.5.4). However, further models have been developed to overcome such difficulties while still using energy minimisation. Variants of active contour models include finite-element snakes [51], snakes with dynamic programming [6, 7], B-snakes [16, 78, 175], snake growing [179], Fourier deformable models [241], polynomial snakes [156, 157], and active shape models [52, 53, 56]. Furthermore, Liu et al. [161] recently combined active contour model principles with the level set method to create parametric-geometric hybrid deformable model, and was applied to medical images.

Active contour models have been used in a wide variety of medical image segmentation applications due to their flexibility [173]. Recently, they have been used to segment cerebrospinal fluid in CT images [202], radiation uptake in PET image [2], and cardiac structures in MRI images [293]. Such recent work [90, 110, 201, 282, 292] indicates that parametric deformable modelling is very much still suitable for numerous medical image analysis tasks [176].

2.5.3 Active Shape Models

Although flexible, parametric contour models can get confused when finding boundary-like areas that are not part of the desired object [173]. To counter this, shape constraints from a statistical model have been used that can correspond to the internal energy, which restricts the degree of flexibility of the deforming model. First proposed by Cootes et al. [52, 53] in 1998, active shape models (ASMs) use a statistical shape model (SSM) [57, 107, 108] based on prior shape knowledge from a training set to restrict the deformation of the model over time. The SSM is based on prior shape knowledge across a set of training shapes, and is built from a set of corresponding landmark points across those shapes. Shape constraints can then be acquired from the SSM, which are applied to the deforming model on every iteration.

The SSM models the distribution of the corresponding points, which essentially statistically models the shape of the object. The amount of variation between corresponding points in the training shapes determines the constraints placed on the model during deformable modelling. A typical ASM scheme would adjust the points of the deformable model based on image-driven forces, then shape constraints are applied to the model which adjusts the points further. This ensures that on every iteration the segmented shape is consistent with the shapes seen already in the training set.

The remainder of this section describes the process of applying SSM constraints on a 2D deformable model [52, 53]. Firstly, some training is required to generate the SSM. Assume a set of n corresponding landmark points across a set of S training meshes. The 2D points on training mesh s can be expressed as;

$$\upsilon_s = (x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n)^T \tag{2.24}$$

It is essential that the set of points across all meshes are globally aligned in order to fairly quantify the variability in shape across the set. Cootes et al. [52, 53] suggests Procrustes analysis [97] for this purpose. Concatenating the globally aligned vectors yields a matrix Υ, and the mean vector $\bar{\upsilon}$ is computed.

To quantify the variation in vectors υ_s, principal component analysis (PCA) is applied to Υ [1, 123]. PCA is essentially a data reduction tool which reduces a large set of variables to a smaller set, but still keeps the majority of information in the larger set. It is a statistical procedure that converts observations which have variables that are possibly correlated, into a set of values that are linearly uncorrelated. The new values are called *principal components*, which highlight the dimensions of highest variability. By computing the covariance matrix and performing eigen-decomposition, a set of eigenvectors and eigenvalues can be obtained [1]. Eigenvectors P represent the dimensional direction of the principal components in the new feature space, while their corresponding eigenvalues λ correspond to how much variation is associated with that principal component (i.e its importance). By multiplying the original observations with the principal components it is possible to transform them to the new feature space of lower dimension, making it easier to deal with data.

The eigenvectors P, eigenvalues λ, and the mean vector $\bar{\upsilon}$ are the components of the SSM. Any shape with a set of n points correspondent to those in Υ can now be expressed in terms of shape parameters b. For every b_i there is a corresponding eigenvector P_i and eigenvalue λ_i. Therefore, during the deformable modelling stage, assume the model has x, y coordinates expressed as;

$$\upsilon = (x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n)^T \tag{2.25}$$

which can be expressed as parameters b using the following expression;

$$b = P^T(\upsilon - \bar{\upsilon}) \tag{2.26}$$

By applying shape constraints on b, we essentially adjust the x, y coordinates of the deforming model υ. The constraints placed on b must ensure that the shape does not exceed the variations of shape seen in Υ. Cootes et al. [52, 53] suggests limits of;

$$b_{i_{min}} = b_i - 3\sqrt{\lambda_i} \tag{2.27}$$

$$b_{i_{max}} = b_i + 3\sqrt{\lambda_i} \tag{2.28}$$

Therefore if any b_i exceed these limits, they must be adjusted accordingly to fall within this range, which yields a new set of shape parameters b_{new}. This can subsequently be translated to point coordinates by obtaining a new vector;

$$\upsilon_{new} = \bar{\upsilon} + Pb_{new} \tag{2.29}$$

Due to its quick implementation and powerful shape regularisation capability, ASM-like methods have been widely used for deformable modelling segmentation [3, 176, 275]. ASM has been used for many medical image applications, such as segmenting brain structures [56, 259], heart structures [56, 64, 196, 267], lungs [259], amongst many others [107]. However, there are a number of aspects in ASM where care must be taken. SSM generation is essential for good segmentation. As Davies et al. [63] explain, the accuracy of the correspondence between points is paramount. Poor correspondence may lead to an SSM that may not adequately describe the shape, and may lead to shape constraint application which allows invalid shape instances. In addition, the amount of point correspondences used to build the SSM must be carefully chosen. An SSM with sparse correspondences may not apply strong enough constraints to ensure shape consistency. However, too many may be too restrictive for segmenting a testing shape instance that has not yet been seen in the training set.

Active appearance models (AAMs) are natural extensions to ASMs, and were first proposed by Cootes et al. [54, 55]. AAMs combine statistical shape models with statistical appearance models, which are generated in a very similar way. Whereas SSMs model the shape variation seen in training shapes, the appearance model also models the texture variation seen in the corresponding training images [55, 58, 243]. This provides even more constraints for a freely deforming model.

2.5.4 Deformable Mesh Models

Active contours have been expanded to 3D while applying energy minimisation [51, 253, 283], and was first implemented with a 3D triangular mesh by Bulpitt et al. [35, 36]. They have become ever more popular recently due to 3D image volumes becoming the norm. 3D mesh deformation can provide smoother resulting segmentations than performing 2D segmentation on a slice-by-slice basis [238]. The same principles apply for mesh deformation, in that one component is used to deform the mesh towards the boundary with image-driven forces, while another component is often used to maintain a regularised mesh surface [127]. Such approaches have been used for soft tissue [109] and heart [197] segmentation, and for the segmentation of bone structures [129, 130], amongst others [181].

Other more recent deformable meshes do not necessarily follow the energy minimisation approach, although the energy components to which they are subject are similar [173, 176]. In these cases the image-driven component and shape regularisation component can be applied separately, sequentially, and in an iterative fashion [127, 288]. For example, the 3D mesh vertices are deformed towards the boundary (usually along the 3D normal direction) using appropriate image features, and is followed by some mesh surface interpolation to provide some shape constraint.

More sophisticated methods have also been incorporated with deformable meshes for boundary detection. Often comparing simplistic features such as gradient, edge or line features are not descriptive enough to move the mesh vertices towards the correct boundary in complex medical volumes. Instead learning-based boundary detectors have been proposed [285,288] where feature vectors from a local patch are classified with machine learning algorithms as being on the boundary or not. This drives the initial deformation based on image data, before shape regularisation in the form of SSM and/or surface interpolation.

A number of modern examples combine such components, and have been successfully used to segment the four chambers of the heart [70, 72, 100, 285], the aorta [120, 288], and the liver [159]. These segmentation procedures were fully automatic, by employing automatic pose estimation. The initial 3D mesh model consisted of both control points and ordinary mesh points. The deformation process started by moving sparse control points towards the boundary using machine learning algorithms, which was followed by simplistic surface interpolation. The remaining points on the surface were then adjusted using the same machine learning classifier. These three steps were iterated to obtain better segmentation results. Examples of such meshes are shown in Figure 2.5.

Furthermore, numerous examples are provided in the literature of deformable meshes coupled with statistical models in order to regularise the mesh shape [64, 107, 108, 180]. In such cases after the mesh vertices have deformed towards the boundary, some soft interpolation is applied before SSM shape constraints are applied, all on an iterative basis. This essentially becomes ASM with deformable meshes, rather than contours, and have shown some impressive results.

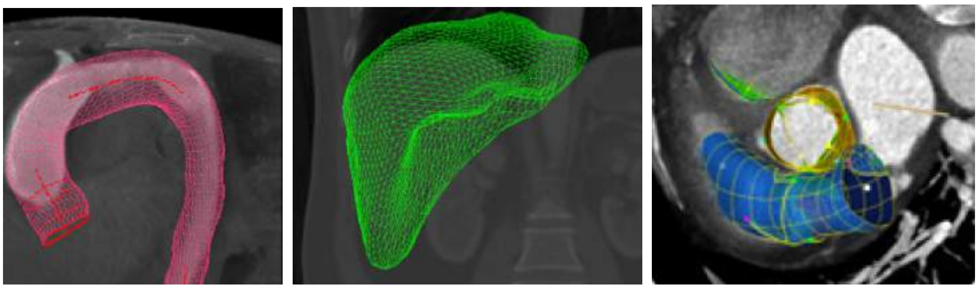


Figure 2.5: Examples of meshes used in medical deformable modelling. Left: Aorta mesh adapted from [285]. Middle: Liver mesh adapted from [159]. Right: Meshes of heart structures adapted from [100].

2.6 Summary

In this chapter necessary background information has been discussed in preparation for presenting the proposed methods in the following chapters. The chapter started by providing an overview on medical imaging, which describes common imaging modalities along with their applications in medicine. This was followed by an overview of conventional approaches for medical image segmentation. Simplistic methods such as intensity, region and classification-based segmentation, were described, as well as more advanced graph-cut segmentation and deformable models. This included a discussion on geometric and parametric deformable modelling.

Intensity-based methods such as thresholding, as well as region-based approaches rely heavily on the assumption that the region of interest is bi-modal. This is unrealistic in medical images, where organ boundaries can appear to be diffused depending on the image modality, and pathological organs tend to have inconsistent intensity values. Classification-based segmentation offers an alternative approach which learns patterns in normal and pathological organs to identify them in an unknown image. However the amount of training data needed is unrealistic for practical medical studies. Graph-cut segmentation is less affected by image noise, however it heavily relies on the source and sink labels, making it ideally suited to interactive, rather than automatic segmentation. Deformable models are known to be well suited to medical image segmentation as they can cope well with structures with high shape variance, which is often the case in pathological organs. Geometric deformable models can also handle topological changes as they are based on curve evolution. However they require a lot of preparation for which level-set and stopping functions to use. Assuming that there aren't any topological changes in valvular or vascular structures, it is thought that parametric models may be more suitable, and the internal and external forces can be weighted and implemented appropriately.

Common parametric deformable model approaches were also presented, which included active contour models or snakes, active shape models and deformable mesh models. Given that the valvular and vascular structures of interest in this thesis are 3D, it is assumed that deformable meshes is the most appropriate approach to take. In addition, active shape models are particularly useful for applying top-down shape constraints on the deformable model, ensuring that the deformation does not yield a shape drastically unlike what we would expect to see.

Finally, an overview of supervised machine learning approaches was presented, which are often used in automatic deformable modelling. Common uses are for the automatic detection of the object for model initialisation, and for boundary detection to drive the model towards the boundary. Descriptions were provided for bagging, boosting and deep learning approaches that are referred to in the following chapters.

The following chapters present an approach to automate SSM generation, which is subsequently used in a proposed image-driven automatic deformable modelling framework. An initial framework is used for the 3D segmentation of the aortic root in CT images. A further development of this framework is also used for vessel segmentation in confocal microscopy images.

Chapter 3

Dense Mesh Correspondence

Contents

3.1 Introduction

This chapter presents a method for automatically establishing dense vertex correspondence across a set of 3D meshes. Such information is used for automatic statistical shape model generation, which is used to provide top-down constraints in deformable modelling. The method is applied to multiple complex aortic root meshes, and is an important component to accelerate model preparation for the segmentation framework presented in Chapter 4.

A common problem in deformable model segmentation is that certain vertices may deform towards the erroneous boundaries that may have strong features. Shape variations can be generalised as a statistical model, which describe the modes of variation in a given shape. Such generalisation can subsequently be imposed on a deformed model in the form of shape constraints [278], resulting in shape preservation. Applying a SSM to a deformed model can ensure that mesh vertices do not deviate too far away from the training shapes, regularising the deformed shape. This essentially means that prior knowledge is provided to constrain the possible shape space. One of the most popular approaches for deformable model based segmentation is through the use of active shape models (ASM), which was first proposed by Cootes et al [52,53]. ASM employs an iterative process where the model is deformed to fit the boundary and is followed by imposing such shape constraints, often leading to better segmentation [54,55,56,57,58].

SSMs can be generated from a set of corresponding landmark points on the shapes' surface, however finding the same number of well-defined, corresponding landmarks in all training shapes is not a trivial task. This is especially true for 3D meshes where the number of faces and vertices can vary. The problem can be even more compounded in anatomical objects where landmark points are often sparsely separated due to their smooth surfaces, where only a few landmarks can be easily identified. It is suggested that building an SSM from sparse landmarks can lead to an SSM that inadequately describes the shape. Such sparse landmarks are useful to characterise regions that are close to these landmark correspondences, but not those that are far away. For example when it comes to medical image segmentation, there is a very real chance that a number of vertices end up on the wrong boundary. If these vertices are far away from the landmarks used to build the SSM, it is possible that applying the SSM has little or no effect on their positions, potentially leaving a shape that does not resemble the desired object at all.

It is therefore proposed that obtaining a denser set of correspondences not only describes the shape better, but also makes it possible to exclude or include landmarks in order to build a SSM of varying degrees of freedom. As such, obtaining a dense set of corresponding landmarks allows flexibility in SSM design. In this chapter, a method is presented for finding correspondences between *all* vertices across a training set of meshes. Any subset of correspondences can then be used to build the SSM. However, to manually segment and label all correspondences across all 3D training shapes is a time-consuming and tedious process, and an automatic method to find shape correspondences is necessary.

A number of approaches to find correspondences on 3D shapes have been proposed, many of which work directly on the meshes themselves [32, 63, 135, 245, 265, 281]. A well known approach for obtaining correspondences for SSM modelling is by using the minimum description length (MDL) [210, 211], proposed by Davies et al. [63]. Here, training shapes are parametrised with a piece-wise linear representation which defines how the points are sampled on each training shape. The shape parametrisations are manipulated according to the optimisation of an objective function, based on the MDL [63, 238, 254]. The approach shows good correspondence between multiple 2D shapes [63, 73, 238], and the approach has been extended to 3D [20,62,63,96]. However, the major drawback of the approach is that it requires numerous function evaluations with stochastic optimisation. For 2D cases alone, Davies et al. [63] state that the method is computationally expensive, and that the method took several hours. It is assumed that this would be far more extreme for finding 3D correspondence. Furthermore, as the method requires good parametrisation, Styner et al. [244] notes that it may perform poorly on complex mesh structures as the optimizer struggles to deal with regions of very high curvature.

Other important approaches in the field include exhaustive correspondence searching [15, 32, 46], or by using distance map representations [121, 135, 219, 265]. Examples include Brett and Taylor's [32] proposed method of expanding an iterative-closest-point (ICP) algorithm for feature point matching. Wang et al. [265], used geodesic distance to match local surface geometry, and similarly Kim et al. [135] used blended intrinsic maps to find non-isometric correspondences in a range of 3D shapes. Although important pieces of work, the methods require water-tight meshes which are not always possible in medical data. This is especially true for aortic root meshes, which are open-ended at the sino-tubular junction. In addition, most of these techniques only establish a set of sparse correspondences, which is not always sufficient to build accurate SSMs of anatomical objects. Furthermore, exhaustive correspondence searching [32], calculating exhaustive representation maps [121,135,219,265], and evaluating numerous parametrisations [62,63], are very computationally expensive. Therefore generally, many current methods are slow and require heavy computation.

Alternatively, non-rigid registration-based approaches have been proposed which are much quicker. In contrast to other methods such as MDL [62,63] and distance map approaches [121,135,219,265] which work in higher-dimensional domains (i.e. parametrisation space and distance map space), registration-based approaches work in the object's spatial domain (being either an image or a mesh). Such approaches generally consist of non-rigidly deforming a source image/mesh to a target, before straightforwardly establishing point correspondence between them.

Zhang et al. [281] used simplistic mesh deformation to find a set of correspondences. The non-rigid transformation was estimated with automatically identified corresponding landmarks of high curvature. This was successfully applied to non-medical meshes such as dinosaur and animal meshes, however identifying such initial landmarks on anatomical structures is difficult as the surfaces are typically smooth.

Other approaches have also been proposed to find correspondences with B-splines, specifically on anatomical shapes. Subsol et al. [245], were among the first to propose using B-splines for non-rigid mesh registration. Line feature matches are found between multiple skull meshes, which are used to estimated a non-rigid transformation between two meshes. After non-rigid registration, point feature matches are found to establish correspondence. However, initial line correspondences are needed here before point correspondences can be established. It is thought that the line features used here may not be suitable for cardiac objects as they tend to have smooth surfaces [82].

Notably, Frangi et al. [81, 82] proposed to find corresponding mesh landmarks by working on the corresponding image data, rather than working directly on the meshes themselves. This method assumes that each anatomical 3D mesh has a corresponding 3D medical image, and that an image atlas and a corresponding atlas mesh has been generated. Desirable sparse landmarks are labelled on the mesh atlas. A non-rigid free-form-deformation (FFD) based on B-splines between each 3D image and an atlas is estimated using an intensity-based similarity metric. This FFD is then applied to the corresponding mesh, where the landmark points are propagated across to establish correspondence. However, using an image-based similarity metric is not always suitable for images that lack strong features or have varying contrast and appearance.

The challenge therefore is to;

- Automatically identify a dense set of corresponding landmarks in order to adequately describe anatomical shapes.
- Not overly rely on noisy cardiac CT image data.
- Develop a method suitable for complex open meshes such as the aortic root.
- Develop a method that is efficient in order to accelerate model training time.

3.2 Contribution & Overview

The idea in this chapter is to obtain a dense set of corresponding landmarks on aortic root meshes. In order to develop an efficient system to accelerate deformable modelling preparation, a non-rigid transformation approach is taken. Following the earlier work of Frangi et al. [82], given a set of sparse corresponding landmarks, this is achieved through; a) simple surface interpolation; b) a new mesh-based similarity metric, and; c) a novel two-stage local transformation.

The method obtains point correspondence between *all* vertices in a target mesh with vertices in a source mesh, and is repeated across the training set. Starting with an initial set of sparsely labelled corresponding landmarks, for simplicity and practicality a denser subset of corresponding landmarks is found through interpolation with Dijkstra's shortest path algorithm [66]. From these landmarks a non-rigid free-form-deformation of source to target is estimated and applied, before complete vertex correspondence is computed between the two meshes. The FFD estimation is similar to that of Frangi et al. [82], however in this case the estimation is based on a similarity metric between the meshes themselves, rather than their corresponding image data.

The proposed method is applied to aortic root meshes generated from CT TAVI images. SSMs of the aortic root have previously been implemented using sparsely separated, easily identified landmarks [100,120]. However a denser set of correspondences is found here to better describe the shape, and to allow flexibility in SSM design. It is shown that the method avoids inaccurate transformation estimation when meshes are generated from CT data, where the appearance of the root can vary significantly due to changing contrast, calcifications, or the blood flow. As demonstrated in this chapter, the technique can find point correspondences among aortic root shapes better than the intensity-based approach taken by Frangi et al. [82].

3.3 Mesh-Based Transformation: Approach

This section describes the idea of finding a dense set of landmark correspondences across a set of triangular training meshes. A target mesh $M_t = (V_t, E_t, F_t)$ with $|V_t| = n$ vertices is selected from the mesh set, and the remaining meshes are regarded as the source meshes. Assume a source mesh $M_s = (V_s, E_s, F_s)$ where

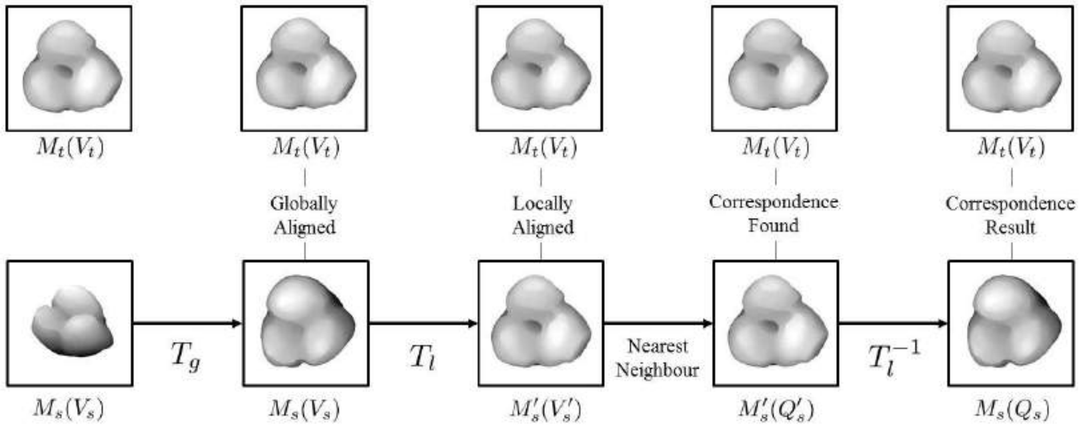


Figure 3.1: Overview of the proposed approach to automatically finding correspondence between two 3D meshes. M_t is the original target mesh with vertices V_t, M_s the original source mesh with vertices V_s, and $M_s'(V_s')$ is the transformed source mesh with vertices V_s'. $M_s'(Q_s')$ is the deformed source mesh, with a reordered set of vertices Q_s' that are correspondent with V_t.

$|V_s| = p$, and $n \neq p$. The challenge now is to find a complete set of n vertex correspondences on the source mesh M_s. The method in this chapter obtains a set of source vertices $Q_s \subset V_s$ that are correspondent with V_t.

Finding correspondences between shapes that vary in appearance can be challenging, however the problem becomes much easier when the shapes are similar to one another. By deforming M_s to match M_t, vertex correspondences can be easily obtained using a simple nearest-neighbour metric.

The source mesh M_s is transformed [225] by $T(x, y, z)$;

$$M_s' = M_s \times T(x, y, z) \tag{3.1}$$

such that $M_s' \mapsto M_t$. To allow flexible matching between the source and target meshes, a local transformation is estimated as well as a global transformation estimation. First the source mesh is transformed to the same global coordinate system as the target mesh. This is followed by a non-rigid transformation estimation between M_s and M_t. Let $T(x, y, z)$ be the transformation that deforms mesh

M_s so that $T : M_s \mapsto M_t$. It is formulated as;

$$T(x, y, z) = T_g(x, y, z) + T_l(x, y, z) \tag{3.2}$$

Following from Frangi et al. [82], alignment to the target coordinate system is estimated through an affine transformation:

$$T_g(x, y, z) = Ax + b \tag{3.3}$$

where A is a matrix containing rotation and scale parameters, and b is a translation vector. These are computed using the ground truth pose parameters of both the source and target meshes.

For the local transformation between M_s and M_t, the FFD is estimated in a multi-resolutional procedure at H resolutions, such that;

$$T_l^H(x, y, z) = \sum_{h=1}^{H} T_l^h(x, y, z) \tag{3.4}$$

To estimate $T_l^h(x, y, z)$, control points $\phi_{i,j,k}^h$ of size $n_x \times n_y \times n_z$ are moved which warp an underlying 3D voxel lattice. At each resolution h the control point spacing is defined as;

$$\delta_h = \delta_0 / 2^h \tag{3.5}$$

where δ_0 is the control point spacing at the original resolution. Decreasing δ_0 decreases the flexibility of the spline, whereas increasing δ_0 allows a more local deformation. Given a set of control points $\phi_{i,j,k}^h$, the transformation is formulated as follows;

$$T_l^h(x, y, z) = \sum_{l=0}^{3} \sum_{m=0}^{3} \sum_{n=0}^{3} B_l(u) B_m(v) B_n(w) \phi_{i+l,j+m,k+n}^h \tag{3.6}$$

where B_l represents the lth basis function of the B-spline. The voxel lattice positions are $i = \lfloor x/n_x \rfloor - 1$, $j = \lfloor y/n_y \rfloor - 1$, and $k = \lfloor z/n_z \rfloor - 1$. $u = x/n_x - \lfloor x/n_x \rfloor$, $v = y/n_y - \lfloor y/n_y \rfloor$, and $w = z/n_z - \lfloor z/n_z \rfloor$ are the fractional positions along the lattice [146, 215, 216].

The position of control points $\phi^h_{i,j,k}$ (i.e. the B-spline parameters) are optimised using gradient descent [237] with the objective function;

$$E(\phi) = E_s(V_t, T_l(V_s)) + \varphi E_r(T_l) \tag{3.7}$$

where E_r is a smoothness cost, φ is a regularisation term, and $T_l(V_s)$ is the deformed V_s given transformation T_l. Finally, E_s is a similarity metric based on the sum-of-squared-distance (SSD) measure [182] between sets of mesh vertices, such that;

$$E_s(V_t, T_l(V_s)) = \frac{1}{|V_t|} \sum (V_t - T_l(V_s))^2 \tag{3.8}$$

where $|V_t|$ is the number of vertices in V_t, i.e L_1 norm.

Establishing point correspondences between two similar shapes now becomes simpler. A simple nearest neighbour algorithm is applied to find complete point correspondence. For *every* vertex in V_t, the nearest neighbour based on Euclidean distance is found in V_s'. The nearest neighbour in V_s' is considered the corresponding vertex, and this leads to a new subset of re-ordered deformed source vertices Q_s'. Finally, the inverse local transformation is applied;

$$Q_s = Q_s' \times T_l^{-1}(x, y, z) \tag{3.9}$$

resulting in the complete set of correspondences Q_s in the source mesh. This is repeated for all source meshes remaining in the set. An overview of the approach is shown in Figure 3.1.

3.4 Application

The method is applied to complex aortic root meshes, which have been labelled from CT TAVI images. For this experiment in addition to both target and source meshes, a sparse subset of corresponding vertices is also used to help the deformation. Therefore we have meshes $M_t = (V_t, E_t, F_t)$ and $M_s = (V_s, E_s, F_s)$, where $|V_t| \neq |V_s|$, as well as sets $P_t \subset V_t$ and $P_s \subset V_s$, where $|P_t| << |V_t|$, $|P_s| << |V_s|$ and $|P_t| = |P_s|$.

Each aortic root mesh was labelled with 10 corresponding landmark points. Three of these were aortic valve *hinge points*; the first of which was the nearest to the *aortic arch*, with the remaining two labelled in a clockwise fashion. Three *commissure points* were labelled between the hinges, with the first commissure point between hinges 1 and 2, and the remaining labelled in a clockwise fashion. Three points were labelled on the *sinotubular junction*, directly below the three hinge points. The first of which was below the first hinge point, with the remaining two once again labelled in a clockwise fashion. Finally, a *centre point* was labelled on the surface at the centre of the root. These points can be reliably identified as they are prominent structural features with high curvature.

Using the 10 landmarks, a denser set of corresponding landmarks were found using an interpolation approach. Pairs of landmark points were defined and the surface paths between them were determined using Dijkstra's shortest path algorithm [66]. Fifteen paths were determined in all; six between the hinge points and their nearest commissure points, three between the hinge points and the centre point, three between the hinge points and their corresponding sinotubular junction points, and three between the sinotubular junction points. Five evenly spaced points were then interpolated along each path, leaving a total of 70 landmark points. Therefore $|P_t| = |P_s| = 70$. An illustration of this can be seen in Figure 3.2.

To make the most of the additional sparse correspondences, this approach estimates the local transformation in two separate stages; a coarser transformation T_{l_1}, followed by a finer transformation T_{l_2}. The full transformation is therefore formulated as;

$$T(x, y, z) = T_g(x, y, z) + T_{l_1}(x, y, z) + T_{l_2}(x, y, z) \tag{3.10}$$

which are applied sequentially. First T_g is applied to M_s before applying both local transformations T_{l_1} and T_{l_2} in sequential fashion;

$$M'_s = M_s \times T_{l_1} \tag{3.11}$$

$$M''_s = M'_s \times T_{l_2} \tag{3.12}$$

so that M''_s is now mapped to M_t. Both T_{l_1} and T_{l_2} are estimated using equations (3.4), (3.5), (3.6), (3.7) and (3.8). A nearest neighbour algorithm is again used,

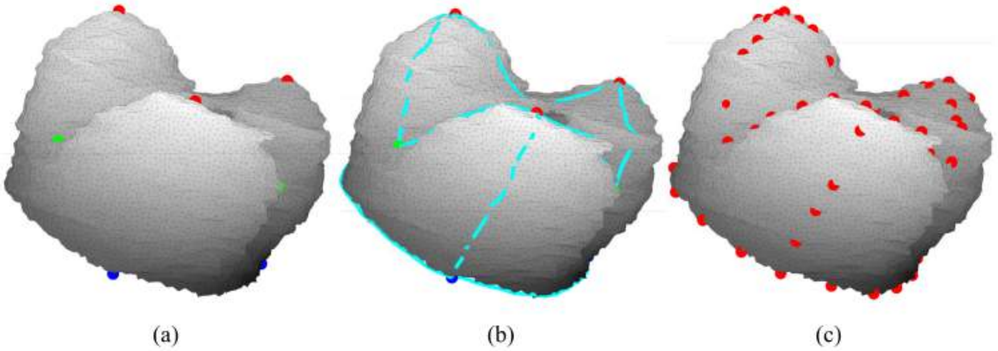


Figure 3.2: Initial landmark labelling. (a) Initial sparse landmark points showing the *hinges* (red), *commissures* (green), and *sinotubular junction points* (blue); (b) Interpolation paths between pairs of initial landmarks; (c) New initial landmarks after path interpolation.

this time to find Q_s'', before the inverse local transformation is used to find Q_s;

$$Q_s = Q_s'' \times (T_{l_1} + T_{l_2})^{-1} \tag{3.13}$$

T_{l_1} is a coarser transformation than T_{l_2}. The intuition is that the first local transformation T_{l_1} provides a better alignment between the overall structure of the two meshes. Once the meshes are aligned, a local transformation T_{l_2} with a higher resolution is used. This alleviates the over-fitting problem, avoids sharp peaks or troughs in the surface of the meshes, while reducing the chance of edge overlapping. Two factors contribute to making T_{l_1} a coarser transformation than T_{l_2}. The first is the similarity metric calculation. For T_{l_1} this is estimated from the SSD between the sparse landmarks P_s and P_t. The similarity metric for T_{l_2} however is based on the SSD between *all* deformed vertices in V_s' and V_t. The other factor is the control point spacing δ_0, during FFD estimation. T_{l_1} has spacing of δ_{0_1}, and T_{l_2} has spacing of δ_{0_2}, where $\delta_{0_1} > \delta_{0_2}$. A relatively large δ_{0_1} is selected as to suppress the amount of local deformation as the FFD was estimated using the sparse set of corresponding landmarks. A smaller δ_{0_2} is used to capture a more local

Transformation	T_{l_1}	T_{l_2}
Similarity Metric	$E_s(P_t, T_{l_1}(P_s))$	$E_s(V_t, T_{l_2}(V_s'))$
Control Point Spacing	$\delta_{0_1} = 15mm$	$\delta_{0_2} = 5mm$
Number of resolutions	$H_1 = 3$	$H_2 = 3$

Table 3.1: Local transformation differences.

deformation by giving the FFD more degrees of freedom. Estimation parameters for T_{l_1} and T_{l_2} are summarised in Table 3.1.

3.5 Comparative Analysis and Results

For comparison, the same framework was implemented, however instead of the proposed mesh-based similarity metric, an image-based metric [82, 215, 216] was used instead. For fair comparison, the 70 *fiducial points* were also used in this implementation, so that the transformation was also formulated as is expressed in Equation 3.10. In addition the same FFD parameters were used. The only difference was the similarity metric. Given that all images are from the same image modality and therefore have similar contrasts, Rueckert et al. [216] state that an SSD between pixel intensities is sufficient.

T_{l_1} was computed in exactly the same way as the proposed method, in order to make use of the provided *fiducial* points. However, instead of using Equation 3.8 to estimate T_{l_2}, an image-based implementation was used instead, such that;

$$E_s(I_t(V_t), T_l(I_s(V_s))) = \frac{1}{|V_t|} \sum (I_t(V_t) - T_l(I_s(V_s)))^2 \tag{3.14}$$

where $I_t(V_t)$ are the voxel intensities in the target image at locations V_t, and $T_l(I_s(V_s))$ is the voxel intensities of the deformed source image at location V_s.

An experiment was conducted using 37 aortic root meshes, where each mesh was used as the target in a leave-one-out fashion. The aortic root, including the ascending aorta and aortic arch were labelled in CT TAVI images, of size

$512 \times 512 \times (500 \sim 800)$, and voxel size was $0.48mm \times 0.48mm \times 0.62mm$. In order to ensure that the hinges were clearly seen, multi-planar-rotation software was used during labelling. This was followed by the marching-cubes algorithm for mesh generation, and the 10 landmarks were manually labelled. This was followed by Dijkstra-based interpolation to yield 70 *fiducial* points on all meshes. The ascending aorta and aortic arch were then discarded from the meshes below the plane on which the 3 sinotubular junction points lay. The average local mesh size was $35mm \times 34mm \times 24mm$.

For the estimation of T_{l_1} it was decided to use $H_1 = 3$ mesh resolutions, and an initial control point spacing of $\delta_0 = 15mm$. A relatively large δ_0 was selected here as to suppress the amount of local deformation as the FFD was estimated using the sparse set of corresponding landmarks. This avoided any sharp peaks or troughs in the surface of the mesh, while also reducing the chance of edge overlapping. $H_2 = 3$ was also used for the estimation of T_{l_2}, however here it was decided to use $\delta_0 = 5mm$ in order to capture a more local deformation by allowing the FFD more degrees of freedom.

Registration of source to target was evaluated using the mean symmetrical point-to-mesh distance $\overline{PMD}$, and symmetrical Hausdorff distance $\overline{HD}$ between M_t and M''_s. It was not possible to quantitatively evaluate the resulting correspondences as no ground truth was provided, and to manually label ~ 8000 corresponding points on all meshes would take an excessive amount of time. For this evaluation it is assumed that better registration between source and target yields better correspondence. Registration results of the proposed method were therefore $\overline{PMD} = 0.4 \pm 0.2mm$ and $\overline{HD} = 3.6 \pm 1.2mm$. This represents a mean distance error of less than one voxel, and a deformed mesh with no distant outlying vertices from the surface.

Table 3.2 compares the registration results of the proposed method with that of Frangi et al. [82]. This shows that both $\overline{PMD}$ and $\overline{HD}$ of the proposed mesh-based metric were dramatically lower than that of the image-based metric. The

Method	$\overline{PMD}$	$\overline{HD}$
Image-based Metric [82]	3.6 ± 4.5	51.4 ± 13.8
Mesh-based Metric	0.4 ± 0.2	3.6 ± 1.2

Table 3.2: Transformation results comparison.

obscured appearance of the boundary between the aortic hinges and the LVOT due to varying blood contrasts has significant effects on the image-based metric. As a result it can be concluded that finding correspondences purely based on image data is not suitable for noisy images such as CT TAVI.

Figure 3.3 compares the deformation of source to target using an image-based metric [82, 216], and the proposed method using the mesh-based approach. This confirms that the proposed method deforms the source meshes so that they have a greater visual resemblance to the target than the deformed meshes using image-based transformation estimation. In addition, the resulting meshes using the proposed method have no compressed or stretched mesh faces, and no tangled mesh edges, leaving a smooth, regularised mesh. This is in contrast to the deformed meshes using an image-based similarity metric, where the meshes appear stretched and compressed in numerous areas.

This close similarity in appearance between M_t and M''_s allowed more accurate corresponding vertices to be found using the nearest-neighbour calculation. Figure 3.4 compares the vertex correspondences using the proposed method, and using the image-based transformation estimation proposed by Frangi et al. [82]. Additional results of vertex correspondences between target and source meshes are shown qualitatively in Figure 3.5. Twenty randomly sampled vertices have been plotted on the target, and the corresponding vertices on two source meshes are shown using a colour coded scheme.

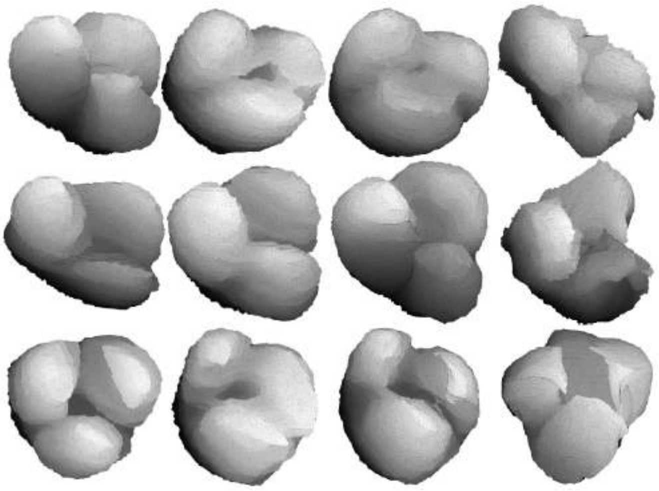


Figure 3.3: Deformation comparison. From left to right; 1st column: Sources (M_s); 2nd column: Target meshes (M_t); 3rd column: Source deformation (M_s'') using proposed method; 4th column: source deformation (M_s'') based on image data [82, 216].

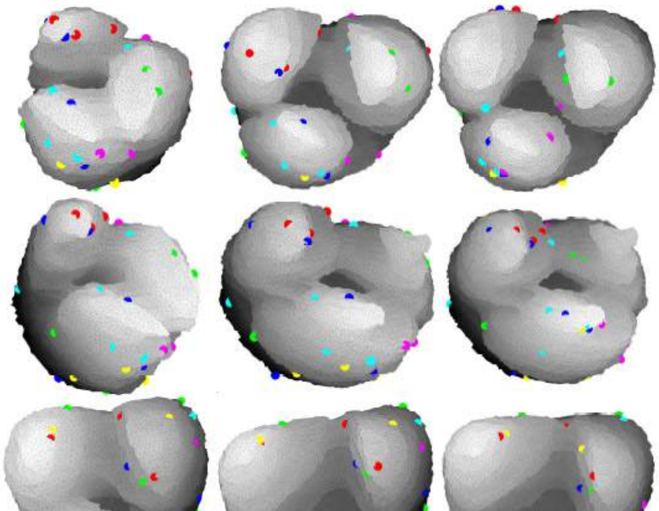


Figure 3.4: Correspondence comparison. From left to right; 1st column: Target; 2nd column: Corresponding vertices using proposed method; 3rd column: Corresponding vertices using image-based method [82, 216].

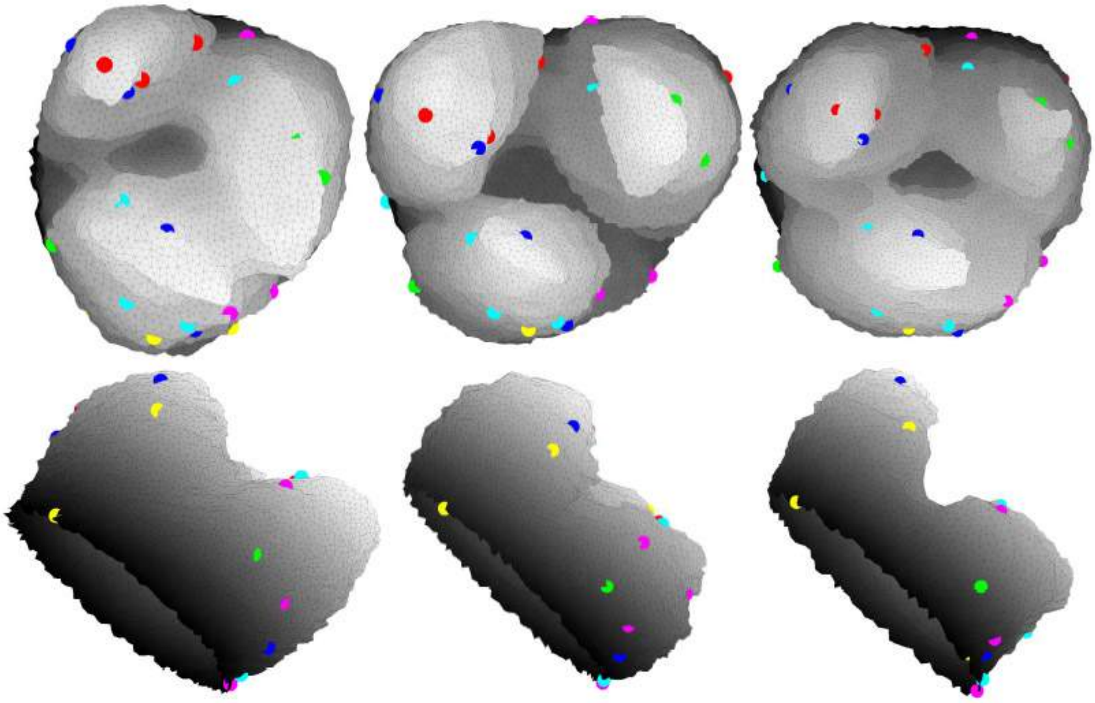


Figure 3.5: Further examples of vertex correspondence using proposed method. From left to right; 1st column: Target mesh with randomly sampled vertices; 2nd column: Randomly selected source mesh with corresponding vertices; 3rd column: Another randomly selected source mesh with corresponding vertices.

3.6 Summary

This chapter presented a mesh based registration method for finding complete vertex correspondence across a set of 3D meshes. In particular, the proposed method has been demonstrated using complex aortic root meshes, which have corresponding images with varying appearance. A set of sparsely distributed corresponding landmarks was used to help with local transformation estimation using a mesh-based similarity metric.

It has been shown that source to target mesh transformation using the mesh-based similarity metric outperformed a similar approach using an image-based metric [82]. Challenging data such as CT TAVI images makes image-based transformation estimation unsuitable due to the highly obscured boundary between the aortic hinges and the left ventricular output tract. Given the images' corresponding meshes and a purely mesh-based metric however, the transformation becomes much more accurate. This allows better correspondence between all mesh vertices across all training meshes.

The next chapter uses the ideas presented in this initial study to automatically generate a statistical model of the aortic root's shape. The statistical shape model (SSM) is used to apply constraints to a deforming model to preserve the shape of the aortic root during segmentation.

Chapter 4

Aortic Root Segmentation

Contents

4.1 Introduction

This chapter presents a parametric deformable modelling framework, for the automatic 3D segmentation of the aortic root in CT TAVI images. A number of studies have proposed segmenting aortic structures in medical images, but very few studies have concentrated on segmenting the aortic root. The majority of studies in the literature have segmented the aortic root as part of a larger segmentation procedure, [64, 72, 100, 262, 288], and most have proposed model-based approaches for this purpose, e.g. [100, 262, 288]. There are non-model-based techniques, but they may easily fail to detect the boundary between the aortic root and the left ventricular output tract (LVOT) due to image noise and calcifications [285, 288]. Model-based approaches intrinsically have some restriction on the shape space, making it more likely to delineate the weak boundary. The remainder of this discussion focusses on model-based approaches.

Waechter et al. [262] proposed an automatic segmentation method that first detects cardiac objects using a generalised Hough transform [116]. This was followed with deforming an initial mesh by formulating an energy function that included a data term based on image appearance and a curvature cost. While they report an extremely low mesh error compared to others in the literature, details of the image resolutions [288] and the complexity of the deformation approach [72] was not reported. In addition, detecting and localising complex anatomical structures, such as the aortic root, with poor image quality using Hough transform in 3D is not only error prone but also computationally expensive due to it's voting system [238]. Bruijne et al. [64] adapted the popular active shape model (ASM) to segment tubular aortic structures. Although the reported mesh error was low, the aortic root was not part of the segmented area in this study. Furthermore, the pose of the initial shape had to be manually placed before segmentation could take place. Elattar et al. [72] applied a 3D normalised cut to segment the aortic root and ascending aorta. Their method produced a low mesh error, however the method was evaluated on a small number of images and 3D graph-cut based methods are computationally expensive.

Notably, recent work by Zheng et al. [288] introduces an aortic root segmentation method that is automatic and efficient. The authors proposed a marginal space learning (MSL) technique to drastically reduce the computational burden of 3D exhaustive search of similarity transform with 9-degree freedom. The position, orientation, and anisotropic scale estimations are carried out incrementally, which leads to sub-optimal solutions but allows efficient search in a much denser computational grid that is otherwise not possible with exhaustive search. The initialised mesh template is then deformed using a learning-based boundary detector and generic mesh smoothing in an iterative process. Grbic et al. [100] adapted this work to segment valvular heart apparatus in 4D. MSL was used for localisation and a multi-linear active shape model, which decomposes the shape space into temporal and spatial components, was used to carry out non-rigid deformation. Both of these methods were very efficient and produced good segmentation results in very little time, however both studies used very large datasets to complete their experiments. Zheng et al. worked with 319 images while Grbic et al. carefully chose their images from an initial dataset of 640 images, making an attempt to recreate these results impractical.

Furthermore, the deformation method in [288] is simplistic and requires the mesh template to be very close to the object of interest after rigid deformation. The movements of individual vertices on the mesh is independent at the non-rigid deformation stage, which can result in highly faceted and unregularised mesh surfaces. Active shape models using high-level constraints defined on a coarse computational mesh can alleviate this problem. However it is predicted that too much reliance on such constraints is often too restrictive for pathological organs, and as is demonstrated in this chapter, it does not allow necessary movement for aortic root segmentation. The challenge therefore is to;

- Develop an efficient deformable modelling framework which provides enough flexibility to converge at difficult boundaries, but also to ensure enough surface regularisation to yield a smooth aortic root mesh.

- Develop a framework with a realistic training set, as often $300 \sim 700$ training images, as has previously been used [100, 288], is impractical.

4.2 Contribution & Overview

In this chapter a fully-automatic, image-driven framework to segment the aortic root in 3D cardiac CT images is proposed. The marginal space learning (MSL) method by Zheng et al. [285, 288] is adopted to efficiently detect and localise the aortic root at natural voxel resolution. Learning-based boundary descriptors are used to drive a free-form-deformation (FFD) with B-spline mesh regularisation. This allows large mesh deformations which is carried out non-iteratively whilst maintaining its mesh regularity, followed by a prior based regularisation using a statistical shape model with thin-plate-spline warping. The proposed framework also allows refined local adjustment to provide flexibility in shape modelling. It is shown that the proposed framework is not only more efficient but also more accurate in segmenting complex anatomical structures with ambiguous image appearance.

Two implementations of the framework is presented. A first attempt at its implementation is closely related to the method proposed by Zheng et al. [285, 288]. However effort is made to maintain a more regularised mesh surface by introducing B-spline mesh regularisation, and alternative features and classification algorithms are used for both MSL and boundary detection. The aim of this is to make the method more computationally efficient compared to [288] by extracting less features and using random forests (RFs). As such, this implementation is referred to as *implementation with random forests*, and uses hand-crafted components for object (RF-MSL) and boundary (RF-BD) detection, as well as manual SSM generation.

This implementation concentrates on developing an efficient process at the testing stage. However, such models require extensive preparation before the segmentation process can begin. For example statistical shape models often need additional manual labelling before their construction, which is both labour intensive and time consuming. In addition, image features are often manually hand-crafted and identified for both boundary detection and pose estimation. Extracting such suitable features is also work intensive, and is often a case of manual trial and error. Another implementation of the proposed framework is therefore presented, with the emphasis on reducing model training time.

The second implementation, referred to as *implementation with neural networks*, uses non-hand-crafted components for both classification and statistical model preparation. This is done by eliminating further manual labelling, through developing an automatic method for identifying landmark correspondences across training shapes. In addition, deep learning-based classification is used for both boundary detection (NN-BD) and pose estimation (NN-MSL), eliminating the need for hand-crafting features. Furthermore, by replacing the RF learning components with NN, preparation time is not only saved, but features are *learned* rather than manually chosen, potentially developing better descriptions for classification.

Section (4.4) describes the overall framework used by both implementations, which are described in Sections (4.5) and (4.6), respectively.

4.3 Dataset

To fully prepare for the TAVI procedure, high quality CT images of the aortic root and aorta are necessary for pre-planning. Once a patient is deemed suitable for the procedure by a cardiologist, clinical staff use cardiac CT images to choose which approach to take, and to decide the size and positioning of the prosthetic valve [93]. Often, cardiac CT imaging has a general acquisition protocol, where the objective is to generate an image of as high a quality as possible of the entire cardiac region. However, as the aortic root and valve are such small structures, slightly different protocols are followed for imaging the aortic region specifically for TAVI pre-planning. This is done by adjusting radiation exposures and acquisition timing. Large datasets have previously been collected and used for medical image analysis [100, 288], however such studies have been carried out by private companies, and the data is not readily available. To our knowledge no public dataset is available where cardiac CT images have been collected specifically to image the aortic root for the TAVI procedure. As such, a dataset was collected in order to test the proposed implementations in this chapter.

Forty six (46) TAVI CT images were collected in collaboration with cardiologists at the Plymouth Hospitals NHS Trust, UK. Of these, 10 were deemed unusable due to inaccurate image acquisition timing, or inconsistent contrast agent distribution, making the aortic root highly obscured. As such 36 image volumes were used throughout the work in this chapter. Each 3D image was of size $512 \times 512 \times (500 \sim 800)$, and the voxel size was $0.48mm \times 0.48mm \times 0.62mm$.

In order to ensure that the valve hinges were clearly seen, multi-planar rotation software was used during image labelling to try and get the best view. Extensive labelling was carried out on all 36 image volumes, with the aortic root, ascending aorta and aortic arch labelled on *all* slices in which they appeared. From these labels, binary volumes were generated, and 3D triangular meshes were subsequently generated using the marching cubes algorithm.

4.4 Image-Driven Segmentation Framework

The proposed deformable modelling framework consists of aligning an initial mesh with an initial aortic root pose estimation, before deforming the mesh towards the object boundary in the image. The initial mesh is the median aortic root mesh from the training set. The estimation of the aortic root pose can be considered as an object detection problem, where the position, orientation, and anisotropic scale parameters need to be determined. An MSL object detection technique [285] is implemented for this purpose (Section 4.4.1).

Once the initial mesh is aligned with the estimated pose, a two-stage mesh deformation method is carried out: non-iterative boundary segmentation followed by iterative boundary refinement. The initial non-iterative process consists of deformable segmentation (Section 4.4.2), which employs a learning-based boundary detector to deform the mesh towards the object boundary, followed by a B-spline-based mesh regularisation technique. Mesh deformations can be large here by defining large boundary detector search paths. This is followed by SSM based regularisation with thin-plate-spline warping (Section 4.4.3), which makes the process capable of large movements without introducing mesh irregularities. The combi-

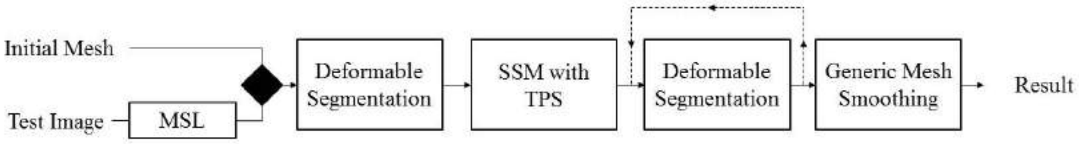


Figure 4.1: Overview of the proposed automatic aortic root segmentation at the testing stage.

nation of these two processes can be considered as an initial segmentation process.

Following initial segmentation, the mesh vertices are locally refined towards the boundary in an iterative fashion to obtain final segmentation. This process involves repeating the deformable segmentation technique (Section 4.4.2) until deformation convergence or the maximum number of iterations is reached. As this process is only finely adjusting the mesh vertices, the boundary detector search path is much smaller here, and decreases with every iteration. The SSM with TPS warping is omitted from this iterative process. Including it would essentially make this an ASM-based method, however it is envisaged that applying strong shape constraints on every iteration may be too restrictive in allowing the mesh to reach the true boundary.

Finally, a generic mesh smoothing technique is used to smooth the mesh surface following the iterative local refinement. The technique proposed by Taubin [248], uses a low-pass filter which removes high curvature variations in the mesh, and is well suited to smoothing faceted mesh faces. Figure 4.1 outlines the steps taken at the testing stage for the automatic segmentation method.

The remainder of this section is as follows. Section (4.4.1) describes the automatic pose estimation with MSL. Section (4.4.2) describes the deformable segmentation technique which is used at both the initial segmentation and local refinement stages. Finally Section (4.4.3) describes the application of shape constraints with SSM.

4.4.1 Marginal Space Learning

The estimation of many pose parameters is often necessary for object detection. In 2D cases for example there may be two parameters for position (x, y), one for orientation θ, and two for scale (S_x, S_y). Conventional exhaustive parameter search comprises of searching every possible combination of position, orientation and scale hypotheses for the desired object, which can be sufficient in 2D. Detection in such cases is often formulated as a classification problem, where the object is either within the patch aligned with a test hypothesis, or not. In 3D however the number of pose parameters increases, for example there may be three for position (x, y, z), three Euler angles for orientation (ω, ϕ, θ), and three for local scale (S_x, S_y, S_z). As a result, to exhaustively search all pose combinations in a single high-dimensional space Ψ is computationally impractical. For example in 3D medical volumes, 9^n hypothesis need to be evaluated, where n is the number of positions to be tested, often in a 512^3 volume. Marginal-space-learning (MSL) is an object detection approach for efficiently testing 3D pose hypotheses which significantly reduces computation compared to exhaustive search. First proposed by Zheng et al. [285], it has successfully been applied to detecting organs in medical images, and in particular to the aortic root in CT images [100, 288].

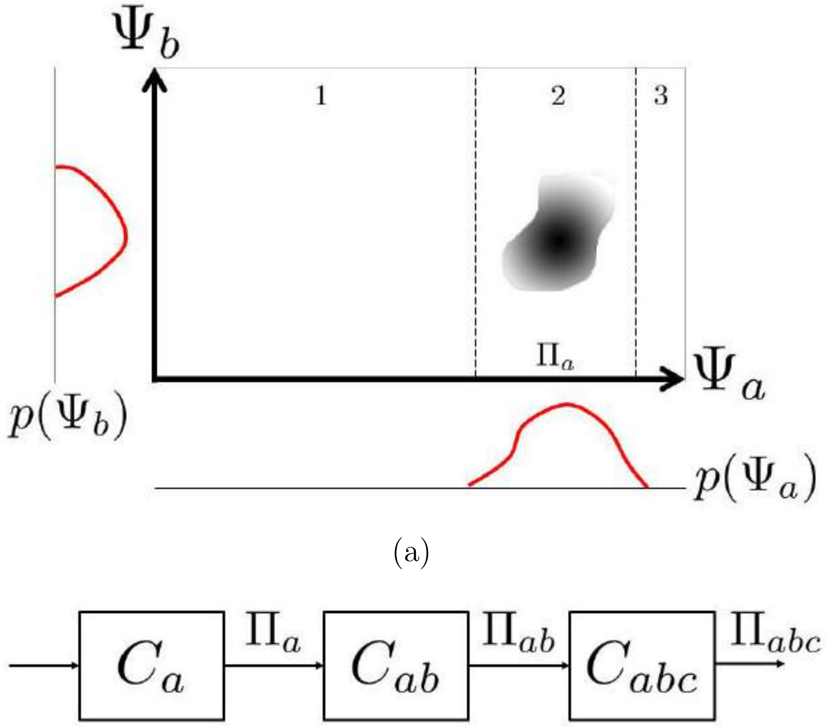
Most anatomical structures have some natural alignment (i.e. the aortic root is always near the left ventricle), therefore it can be observed that the probability distribution is clustered in a small region of Ψ. The idea of MSL is that the full similarity search space can be marginalised;

$$\Psi_a \subset \Psi_{ab} \subset \Psi_{abc} = \Psi \tag{4.1}$$

where Ψ_a is the position search space, Ψ_{ab} is the position-orientation space, and Ψ_{abc} is the position-orientation-scale space. Assuming some natural alignment, it is assumed that the optimal pose hypothesis Π is contained within the highest probability hypotheses of all marginal spaces, such that;

$$\Pi = \Pi_{abc} \subset \Pi_{ab} \subset \Pi_a \tag{4.2}$$

For illustration, consider a 2D search space as illustrated in Figure 4.2 (a). Here it can be seen that the hypotheses Π_a with the highest detection probability $p(\Psi_a)$



(a)

(b)

Figure 4.2: a) An illustration of MSL searching in 2D. b) An illustration of 3D MSL. Conventionally C_a is the position estimator which finds high scoring (x, y, z) hypotheses, Π_a. C_{ab} finds the highest scoring $(x, y, z, \omega, \phi, \theta)$ hypotheses Π_{ab}, and finally C_{abc} finds the highest scoring $(x, y, z, \omega, \phi, \theta, S_x, S_y, S_z)$ hypothesis Π_{abc}.

in Ψ_a are clustered in region 2. When finding the highest scoring hypotheses Π_{ab} based on the joint probability $p(\Psi_a, \Psi_b)$, it now becomes obvious that considering hypotheses in regions 1 and 3 become unnecessary, wasting valuable computation and time. Therefore by searching in marginal space Ψ_a first, the vast majority of high-dimensional hypotheses can be eliminated.

In 3D it is assumed that three separate pose classifiers can be used for object detection, instead of a single unified one used in exhaustive search. Given the three marginal spaces in (4.1), three classifiers C_a, C_{ab} and C_{abc} can be trained. At the testing stage C_a can eliminate the vast majority of false hypotheses in Ψ_a, leaving high probability hypotheses Π_a. These are then passed through C_{ab}

to leave Π_{ab}, which are subsequently passed through C_{abc} to leave $\Pi_{abc} = \Pi$. A schematic diagram is shown in Figure 4.2(b). In reality the last stage is an exhaustive search but with very few hypotheses to evaluate. MSL therefore dramatically alleviates the high computation needed for exhaustive search by significantly reducing the number of high-dimensional hypotheses to test. Furthermore, MSL has been shown to reduce the number of test hypotheses by 10^6 [285].

For this reason, an MSL parameter search approach is implemented for the aortic root detection problem. The efficiency is increased further by training just two MSL classifiers rather than three, and reducing the number of features used for the two classifiers. A position estimator C_a tests all position hypotheses in the volume. The highest scoring (x, y, z) hypotheses Π_a are then passed through a position-orientation estimator C_{ab}. In this implementation the highest scoring $(x, y, z, \omega, \phi, \theta)$ hypothesis is then taken as the position-orientation result. Finally for simplicity the mean local scale of the training meshes is used to yield a 9-element pose estimation vector $(x, y, z, \omega, \phi, \theta, S_x, S_y, S_z)$. This assumption is made as there is a relatively low variance in scale amongst the training shapes. A schematic diagram of the MSL pipeline is shown in Figure 4.3.

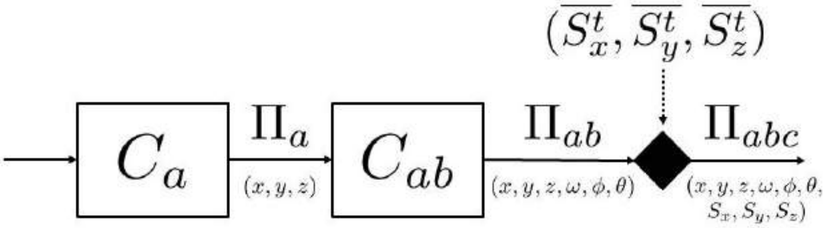

$(\overline{S_x^t}, \overline{S_y^t}, \overline{S_z^t})$
C_a
Π_a
(x, y, z)
C_{ab}
Π_{ab}
$(x, y, z, \omega, \phi, \theta)$
Π_{abc}
$(x, y, z, \omega, \phi, \theta, S_x, S_y, S_z)$

Figure 4.3: A schematic illustration of the 3D MSL implementation. The highest scoring (x, y, z) hypotheses Π_a is obtained from C_a, and the highest scoring $(x, y, z, \omega, \phi, \theta)$ hypothesis Π_{ab} is obtained from C_{ab}. For efficiency the mean local scale of the training set is included to complete the full-similarity estimation $(x, y, z, \omega, \phi, \theta, S_x, S_y, S_z)$.

During training, a positive hypothesis must satisfy the following condition for all k parameters;

$$|P_k - P_k^t|/S_k \ \leq 1 \forall k \tag{4.3}$$

where P_k is a single pose parameter hypothesis, P_k^t is its corresponding ground truth, and S_k is the corresponding parameter search step. For the position estimator, $P = (x, y, z)$, $P^t = (x_t, y_t, z_t)$, and $S = (1, 1, 1)$ voxels. For the position-orientation estimator, $P = (x, y, z, \omega, \phi, \theta)$, $P^t = (x_t, y_t, z_t, \omega_t, \phi_t, \theta_t)$, and $S = (1, 1, 1, 10°, 10°, 10°)$.

4.4.2 Deformable Segmentation

A learning-based boundary detector is applied to localise the aortic root boundary. Once the initial mesh is aligned with the estimated pose, the mesh vertices must be adjusted to fit the object boundary. To avoid vertex entanglement, a search path is defined along the surface normal direction for each vertex. The path coordinate that has the strongest boundary response is then taken as the new vertex position.

After boundary detection, we have a set of original mesh vertices V, and a new set of vertex positions V' that are on the boundary. Due to the length of the search path and image noise, it is possible that a vertex may have found an optimal boundary response further down the search path than its neighbouring vertex. This can lead to vertices crossing over, resulting in tangled and overlapping mesh faces. To address this problem, a B-spline based mesh regularisation method is proposed which estimates a non-rigid transformation between V and V' before performing a free-form-deformation (FFD) [225] on V to fit V'.

The FFD is estimated by warping an underlying voxel lattice controlled by a set of control points which act as parameters of a B-spline. The control points are defined as $\phi^h_{i,j,k}$ of size $n_x \times n_y \times n_z$, which are separated by δ, and the FFD is formulated as follows;

$$T(x, y, z) = \sum_{l=0}^{3} \sum_{m=0}^{3} \sum_{n=0}^{3} B_l(u) B_m(v) B_n(w) \phi_{i+l,j+m,k+n} \tag{4.4}$$

where B_l represents the lth basis function of the B-spline. The voxel lattice positions are $i = \lfloor x/n_x \rfloor - 1$, $j = \lfloor y/n_y \rfloor - 1$, and $k = \lfloor z/n_z \rfloor - 1$. $u = x/n_x - \lfloor x/n_x \rfloor$,

$v = y/n_y - \lfloor y/n_y \rfloor$, and $w = z/n_z - \lfloor z/n_z \rfloor$ are the fractional positions along the lattice [146, 216].

The control point spacing δ has a direct influence on the degrees of freedom allowed for the B-spline. A low δ value results in a more localised deformation, raising the potential for V to deform more closely to V', leaving tangled mesh edges still. A high δ value allows a more global deformation as the control points are more sparsely spread, reducing the chances of tangling the mesh edges, but resulting in V deforming further away from the boundary. A trade-off must be found to allow V to deform as close to the boundary positions as possible, V', without overlapping each other.

In addition, the non-rigid transformation is estimated in a multi-resolution procedure which is expressed as a summation of FFDs at multiple resolutions H [146, 216].

$$T^H(x, y, z) = \sum_{h=1}^{H} T^h(x, y, z) \tag{4.5}$$

At each mesh resolution h, the voxel lattice is warped by moving the set of control points $\phi^h_{i,j,k}$ which is consequential of δ_h, computed as;

$$\delta_h = \delta_0 / 2^h \tag{4.6}$$

where δ_0 is the original control point spacing and h is the resolution level. The B-spline parameters $\phi^h_{i,j,k}$, are then optimised using gradient descent [237], formulated as

$$E(\phi) = E_s(V', V) + \varphi E_r(T), \tag{4.7}$$

where E_r is a smoothness cost and φ is a constant that defines the contribution of the smoothness term. E_s is a similarity metric, which is a sum-of-squared-difference (SSD) metric between V and V', such that;

$$E_s(V', V) = \frac{1}{n} \sum (V' - T(V))^2 \tag{4.8}$$

Where n is the number of vertices in V, and $T(V)$ is the deformed initial mesh. For these experiments the number of mesh resolutions during transformation estimation is set as $H = 3$, and the FFD control point spacing $\delta_0 = 10$. An example of the effect of the mesh regularisation is shown if Figure 4.4.

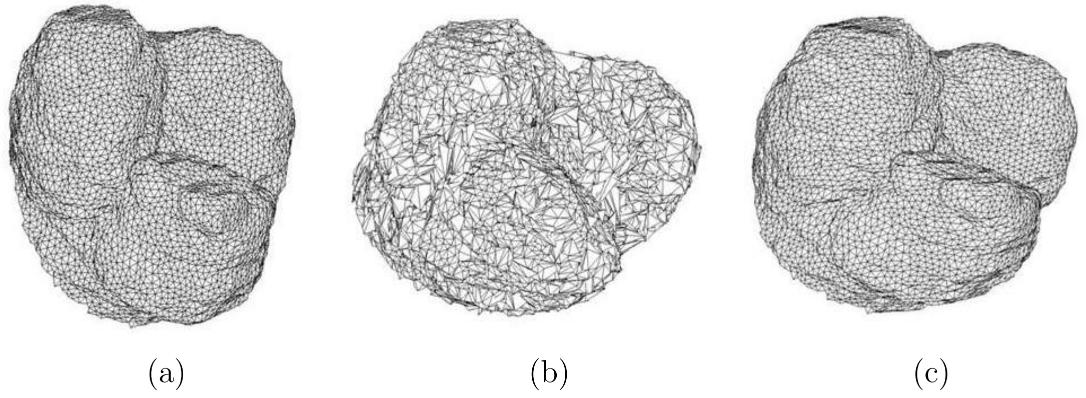


Figure 4.4: a) Original mesh with vertices V. b) Deformed mesh after boundary detection, with vertices V'. c) Deformed mesh after boundary detection and mesh regularisation.

4.4.3 SSM & Shape Constraints

Having performed mesh deformations that are capable of large movements, the shape space needs to be constrained so that any deformation is consistent with the shapes in the training set. Applying constraints based on statistical-shape-models (SSMs) is a popular approach to provide prior shape knowledge to a deforming model, and was first proposed by Cootes et al. with active-shape-models [53]. It is believed that an SSM built with a complete set of vertices may be too restrictive during mesh deformation, however too few number of correspondences may not adequately describe the shape. Therefore an SSM with a corresponding subset of mesh vertices called *fiducial* points, is built here. The SSM is built and the constraints are applied consistent with Cootes et al. [52, 53] , however it is worth noting that the constraints are not applied in an iterative fashion, unlike ASM.

To generate a 3D SSM, a set of n corresponding landmark points are obtained across a training set of meshes. A set of 3D point coordinates on a single mesh can be expressed as;

$$v_s = (x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n, z_1, z_2, \ldots, z_n)^T \tag{4.9}$$

S such corresponding vectors are also obtained across S training meshes, yielding a $3n \times S$ matrix Υ, on which principal component analysis (PCA) [123] is applied. This yields a set of mean points $\bar{\upsilon}$, as well as a set of eigenvectors P from the covariance matrix and their corresponding eigenvalues λ. P describes the modes of variation between the sets of points υ_s in Υ, and λ corresponds to their variance. In an attempt to reduce the noise in the variations in Υ, only eigenvectors which show up to 98% variance are kept. This yields a set of eigenvectors $P = (p_1|p_2|\ldots|p_t)$ and a corresponding set of eigenvalues $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_t)$. These parameters along with $\bar{\upsilon}$ are the main components of the SSM which are used to apply the shape constraints.

To apply the shape constraints after deformable segmentation, the mesh's corresponding (x, y, z) *fiducial* points are rearranged such that;

$$\upsilon = (x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n, z_1, z_2, \ldots, z_n)^T \tag{4.10}$$

Using the SSM parameters, this can be represented as a set of deformable model parameters $b = (b_1, b_2, \ldots, b_t)$, which is formulated as;

$$b = P^T(\upsilon - \bar{\upsilon}) \tag{4.11}$$

Therefore by adjusting the values of elements in b it is possible to adjust the positions of υ. The variance of each element in $b = (b_1, b_2, \ldots, b_t)$ is given by the corresponding elements in $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_t)$. As a result by adjusting every element of b so that they do not exceed a limit of;

$$b_{i_{max}} = m\sqrt{\lambda_i} \tag{4.12}$$

where $m = 3$, the model parameters become similar to the model parameters of the shapes in the training set. This yields a new set of parameters b'. Finally, the constraints placed on b can be translated to constraints on the coordinates υ using;

$$\upsilon' = \bar{\upsilon} + Pb' \tag{4.13}$$

After applying the SSM shape constraints, only the *fiducial* points υ are curbed to υ', therefore a further process is required to adjust the remaining vertices appropriately. Thin-plate-spline (TPS) warping [21] allows surface interpolation depending on the physical bending energy of a thin metal plate, whilst keeping certain nodes fixed in location. This makes it ideal for fixing the adjusted *fiducial* points in place while interpolating the remaining vertices around them. Using SSM constraints in conjunction with TPS effectively allows shape constraint application to the entire mesh.

4.5 Implementation with Random Forests

The framework uses components such as MSL and learning-based boundary detection which are closely related to Zheng et al. [285]. However, the number of features extracted from the MSL position estimator, position-orientation estimator, and the boundary detector are very large. This makes feature extraction not only time and computationally consuming at the training stage, but crucially also at the testing stage. Furthermore, boosting algorithms are also used for classification in these components, which can be another computationally consuming processes.

Given that the ultimate goal of deformable modelling is to produce accurate segmentation results in real-time, in this section the aim is to make these components more computationally efficient. Features are specifically chosen so that the feature vector length is dramatically reduced compared to Zheng et al. [288], and random forest algorithms are used for classification.

4.5.1 Pose Estimation: RF-MSL

Conventional MSL implementations [100, 285, 288] extract exhaustive wavelet-like Haar features [261] for the position estimator and is trained with a boosting algorithm [257]. Similarly, thousands of *steerable features* have been extracted for the position-orientation estimator [285]. Such feature extraction however can produce a feature vector length of hundreds of thousands due to the large patch size which should be similar to the global scale of the desired object, making feature extraction impractical and inefficient. The proposed implementation attempts to increase efficiency by significantly reducing the number of features, and to train the estimators using random forests (RFs). This is referred to as RF-MSL.

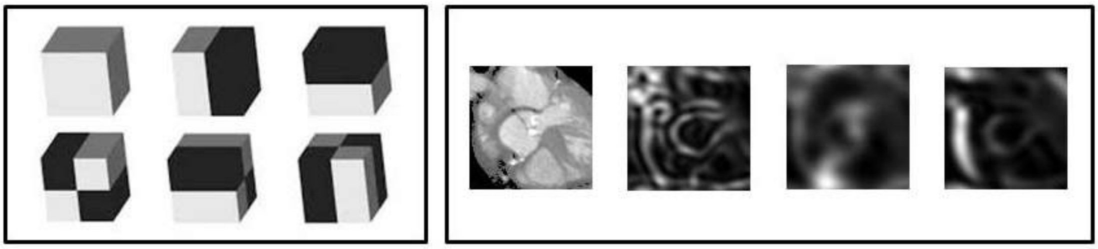


Figure 4.5: Left: Example 3D Haar features (adapted from [126]). Right example SHOG features maps around the aortic root.

For the position estimator, coarse Haar features at a number of scales ranging in size between 0.25-3 times the mean global scale of the training meshes $(\overline{Sx_t^g}, \overline{Sy_t^g}, \overline{Sz_t^g})$, are extracted. These are centred at the position hypothesis, and incorporate large appearance differences around the root, as well as local appearance. Using these features alone is sufficient for finding the tubular structure of the aorta, however additional features are required to successfully detect the aortic root. For this purpose, 3D spherical HOG (SHOG) features [160, 235] are extracted, which highlight areas of high curvature. These areas are heavily present around the aortic hinges in the root, and are not present along the tubular structures of the ascending aorta and aortic arch. An illustration of such features are shown in Figure 4.5. This makes these features substantially more discriminative with regards to the aortic root. A total of 358 features are extracted for the position estimator.

Similarly, coarse Haar and SHOG features are also extracted for the position-orientation estimator. However, whereas the position estimator features are extracted from a large patch aligned with the global axes, the position-orientation estimator features are extracted from a patch aligned with the orientation hypothesis. As the detailed region of the root has already been identified from the previous estimator, even coarser features are extracted to estimate its orientation, which takes into consideration the direction of large structures such as the ascending aorta and aortic arch. To this end, a total of 63 corse features are extracted.

Both estimators are trained with random forests due to its ability to handle a large number of training samples, and that parameter tuning is straightforward, essentially meaning that it works “out of the box”. In addition, boosting algorithms tend to be time consuming and computationally expensive to train. Both RF-MSL classifiers were trained with a positive/negative sample ratio of $\frac{1}{5}$, with all possible positive samples (i.e object) samples being used. Both random forests were trained with 200 trees, which yielded an output voting score for each test sample.

To evaluate the performance of the pose estimation, the Euclidean distances of the (x, y, z) voxel coordinates between the ground truth and estimation was computed. This produced an average error of 5.6 ± 2.9 voxels. Similarly the mean absolute difference of the (ω, ϕ, θ) Euler angles were computed. This produced an average error of 13.8 degrees.

4.5.2 Boundary Detection: RF-BD

An increase in efficiency is attempted for the aortic root boundary detector compared to previous implementations [100, 288] by extracting just one 2-rectangle Haar feature [261], and using random forest classification. This is referred to as RF-BD.

The vertex normals can be straightforwardly determined from the mesh, therefore the Haar templates can be aligned with the normal direction. As a positive sample is a ground truth vertex sitting on the object boundary, one rectangle box will be inside the object whilst the other will be outside, yielding an optimal boundary response. This feature, along with the mean intensity of each rectangle yields a total of just three features.

The Haar feature is extracted at a small scale, with the feature being $3 \times 3 \times 3$ voxels in size. The reason for extracting a Haar feature of such a small size is that in many instances, the boundaries of the hinges of the aortic root and the left ventricular output tract (LVOT) are highly diffused. Furthermore if the LVOT is filled with blood flow, the boundaries may be within extremely close proximity to one another. However this is not the case in all image instances, and is dependent on whether the image was taken during cardiac systole or diastole. As a result if larger patches are extracted there is a high likelihood that many boundary patches may be inconsistent, making training a more difficult challenge. Extracting small patches aims to alleviate this problem.

RF was used to train the detector for the reason that it is easy to tune, and can easily handle a large number of training samples. The detector was trained with 100,000 training samples and 200 trees, which produced a detector with boundary sensitivity of 54.5% and specificity of 88.4%. Low boundary sensitivity would be expected as detecting the boundary between the aortic hinges and the left ventricular output tract is a difficult challenge. The blood flow between the aortic valve and the left ventricle makes this region highly diffused, therefore it is expected that boundary classification errors would be seen here. The specificity is high, meaning that the majority of non-boundary voxels were correctly classified. Furthermore, the vast majority of false-positive classifications (i.e incorrectly classified non-boundary voxels) were in close proximity to boundary ground truth, with a mean Euclidean distance of 5.1 voxels. A histogram is shown in Figure 4.6 of the false-positive counts versus the Euclidean distance between the false-positive voxel and its nearest ground truth boundary voxel. As a result the RF-BD is sufficient for this framework, where additional components such as shape and mesh regularisation are employed to help the segmentation.

4.5.3 Manual Correspondence for SSM

To generate a SSM, a set of corresponding surface points were manually labelled on all meshes in the training set. This essentially follows exactly the same process used for labelling the *initial* landmark points in Chapter 3. Each training mesh is labelled with 10 landmarks on distinctive anatomical features, and a further 60 points were interpolated between them to yield 70 corresponding *fiducial* points across all meshes. Figure 3.2 (Chapter 3) illustrates this.

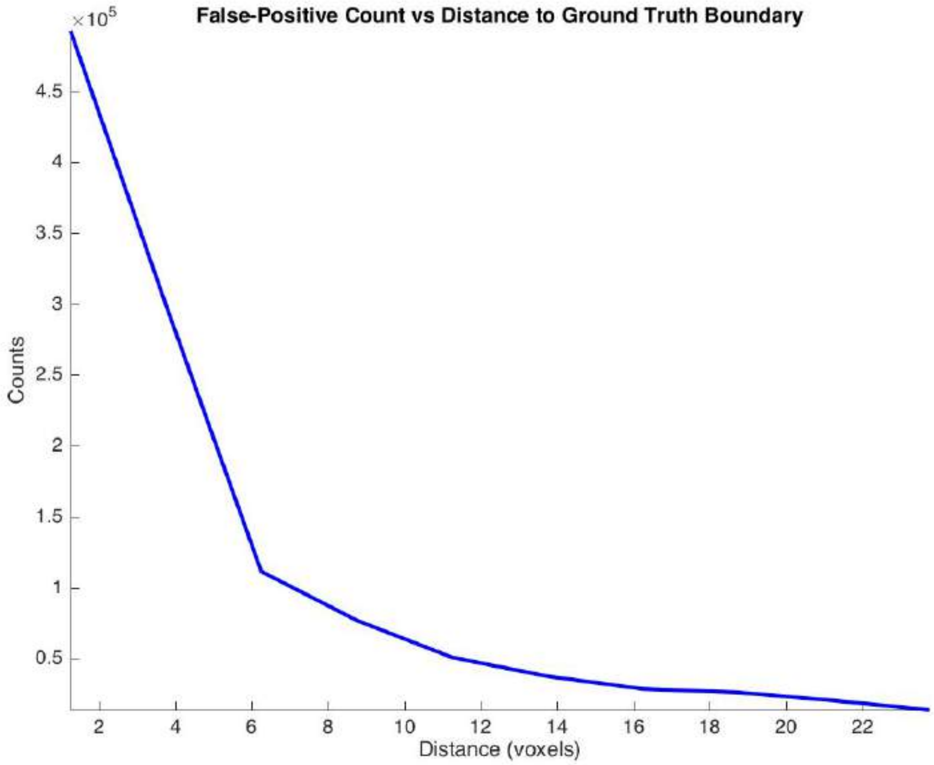


Figure 4.6: Histogram of nearest neighbour distances with RF-BD.

4.6 Implementation with Neural Networks

In the previous Section (4.5), the aim was to implement an efficient deformable modelling framework which accurately delineates the image boundary while maintaining a regularised mesh surface. However, supervised automatic 3D deformable modelling is not only computationally demanding during testing, but it is also labour intensive in preparing training data. Parameters for effective model regularisation as well as useful feature extraction are chosen carefully depending on the application, which can be extremely time consuming. In this Section the aim is to alleviate the burden of lengthy training processes and excessive manual work, using the same general deformable modelling framework as was previously used. The challenges involved with accelerating model preparation and reducing manual work is three-fold. 1) Building an SSM often requires manual labelling for identifying landmark correspondences across the training set. 2) Potentially complex features capable of estimating object position and orientation are often manually hand-crafted. 3) Boundary features are also often hand-picked, resulting in time-consuming processes.

Model regularisation regularly requires building a statistical model [53] which often demands additional manual labelling. As was outlined in Chapter 3, appropriate automatic solutions to building statistical models are not well reported in the literature. However the technique used in Chapter 3 is used for this purpose here, which eliminates the need for additional manual labelling.

Similarly, choosing optimised discriminative features for both object and boundary detection can be excessively lengthy processes. Selecting appropriate features for both the pose estimation classifiers as well as the boundary detector can be challenging as feature type, orientation and scale must be considered, and pathological structures often look significantly different to each other. The argument here is that deep learning based approaches should be adopted.

Deep-learning algorithms are popular alternatives as they are capable of learning features by combining inputs and adjusting their importance [143]. Recently there has been tremendous work in their application to medical image analysis [122], and in particular CNNs for anatomical organ detection [233]. Given that the appearance of local pathological structures significantly varies, global structures are often just as important for detection as local structures. However incorporating them requires training with large patches, which makes applying CNNs to 3D medical data computationally impractical. Instead it is proposed that deep learning is used by combining NNs with novel multi-resolution pooling to learn features from raw intensities, without the need for convolution operations. This reduces the number of network inputs, simplifying the network so that the number of weighting parameters to be optimised is dramatically decreased. This is integrated with MSL, which not only eliminates the burden of defining hand-crafted features during training, but it also significantly reduces the number of pose hypotheses at the testing stage. This is referred to as NN-MSL.

Related work in this area has been carried out by Ghesu et al. [94], who combined deep learning with MSL for object detection in ultrasound images. Their method is referred to as marginal-space-deep-learning (MSDL), and raw intensity features are inputted into a large neural network for feature learning. It is envisaged that intensity information is required from both large-scale global structures, as well as smaller, more local structures for aortic root detection, yielding a large number of network input nodes. Such an approach has a major challenge to overcome however. That is how to reduce the number of parameters to optimise in order to accelerate the NN training process? Notably, Ghesu et al. [94] use an additional learning cascade for network simplification, which aims to reduce the number of nodes in the network. However despite this, their MSDL networks are still very large, with over 1.2 million parameter weights and biases needing to be optimised, from 5832 inputs for sufficient performance. It is therefore assumed that such large network architectures are very time consuming during training. To remedy this, multi-resolution pooling is proposed here, which incorporates information from both global and local structures. The proposed NN-MSL has a combined total of ∼43,000 parameters from 387 inputs, which is ∼30 times less than MSDL. The proposed architectures therefore requires significantly less computation and dramatically speeds up the training time.

Therefore in this section an automatic deformable modelling technique is presented that applies efficient deep learning algorithms and low level regularisation to avoid over-constraint and ensure adaptability. The work differs from the previous section in three ways; 1) MSL is adapted by training NNs (NN-MSL) with multi-resolution pooling capable of *learning* useful features; 2) a NN is also used for boundary detection (NN-BD) which learns features capable of discriminating between boundary and non-boundary voxels; 3) The time to generate an SSM is significantly reduced by automatically finding point correspondences across the training set.

4.6.1 Pose Estimation: NN-MSL

Given that pathological anatomical structures can significantly vary in appearance, defining hand-crafted features suitable for detection can be difficult. Deep-learning algorithms are employed to train the MSL classifiers as they are capable of learning features from a set of raw voxel intensities.

Global structures such as lungs and heart chambers are important for distinguishing between different valve-like structures such as the aortic and pulmonary valves. However extracting large patch intensities leads to large feature vectors, making the process computationally expensive. Sampling at lower resolutions eases this at the risk of losing local structures highlighting the valve in the first instance. To avoid this, multi-resolution pooling on an irregular grid is introduced. Given a large patch of intensities, multi-resolution pooling decreases in resolution from the centre out. This leaves dense patch coordinates at the centre and increasingly sparse coordinates towards the perimeter, yielding an irregular grid, making convolution operations, and therefore CNNs infeasible. As a result the NNs used here are built as follows. 1) Intensities from large patches correspond to the network inputs; 2) Multi-resolution pooling, yields a significantly reduced feature vector on an irregular grid; 3) Pooled patch values are fed into several fully-connected layers. An example is shown in Fig.4.7.

Multilayer NNs are composed of layers where nodes in adjacent layers are linked together with weighted connections [103, 143]. Node i in layer k is connected to node j in layer $k+1$ by weight w_{ji}^{k}, and is given a bias term b_{j}^{k+1}. The activation of a node at layer k is computed and used as an input for the nodes in layer $k+1$, and is repeated until the output activations are computed. An activation a_{j}^{k+1} is computed by calculating the weighted sum of the node's inputs, and passing it through a sigmoid function f, formulated as

$$a_{j}^{k+1} = f(b_{j}^{k+1} + \sum_{i=1} \mu_{i}^{k} w_{ji}^{k}) \tag{4.14}$$

where μ_{i}^{k} is the node's input. The network therefore fits a function to a supervised output, where w_{ji}^{k} and b_{j}^{k+1} are optimised using the *Levenberg-Marquardt*

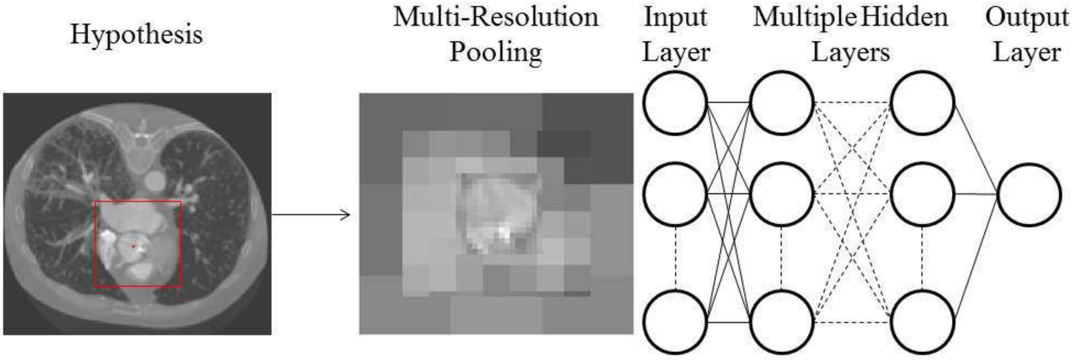


Figure 4.7: Illustration of proposed multi-layer NN with multi-resolution pooling.

[152, 171] and *back-propagation* [218] algorithms, meaning the network is capable of learning useful features. Increasing the number of nodes in a layer increases the number that can be combined for a node in the next layer. Increasing the number of layers in the network increases the complexity of node combinations, leading to ever abstract features. Too few layers, and nodes in layers can lead to a network that is too generalised, whereas too many can easily lead to overfitting the data. A balance between the two must be found depending on the data used and its application.

The NN-MSL position estimator consisted of a patch size 3 times the mean global scale, and was pooled at resolutions of $[\frac{1}{8}, \frac{1}{40}, \frac{1}{56}]$ times the original resolution. This yielded 387 inputs. The NN had 3 hidden layers of $[50, 25, 10]$ nodes, and samples were selected with an object/non-object ratio of $\frac{1}{50}$. This resulted in a mean position estimation error of 12.2 ± 4.2 voxels at full resolution. Position-orientation inputs were taken from a patch 4 times the size of the mean local scale and pooled at resolutions of $[\frac{1}{8}, \frac{1}{40}, \frac{1}{96}]$, yielding ~ 500 inputs. Two layers of $[50, 25]$ nodes were used with an object/non-object ratio of $\frac{1}{25}$. Its mean error was 15.4 degrees, equating to ~ 1.5 search steps. The number of layers in each network were selected to ensure sufficiently abstract features were learned.

4.6.2 Boundary Detection: NN-BD

Learning-based boundary detectors have been implemented for detecting the aortic root boundary using hand-crafted features [285,288], however efficiently determining such features can also be challenging. Here a NN similar to the that in Section (4.6.1) is used. The NN-BD learns features from a set of intensities of a local patch centred at the search path coordinate, and aligned with the direction of the path. For similar reasons to the RF-BD in Section (4.5.2), it is necessary for a small local patch to be extracted here. The blood flow between the aortic valve and the left ventricle is inconsistent across the image instances, therefore a small patch highlighting the narrow boundary must be used.

Convolutional neural networks (CNNs) are highly suitable for image classification [143,206,220] as the features learned here are spatially invariant. However, to perform convolution at possibly multiple layers, the original image (local patch in this case) must be of sufficient size, as the layer output decreases with every convolutional operator. Given that a small local patch is necessary in this case, it is infeasible to design a conventional CNN with multiple convolutional layers, and designing such a network is out of the scope of this thesis. To this end, a NN is used where features are learned from nodes that are spatially variant.

The NN input nodes therefore corresponded to intensities that were extracted from a $3 \times 3 \times 3$ local patch, which corresponds to 27 nodes. This is followed by a single hidden layer of 10 nodes which was sufficient for boundary detection. The network was trained with an equal boundary/non-boundary ratio and trained with a total of 50,000 samples. This resulted in a detector with boundary sensitivity of 66.2%, and sensitivity of 86.5%. In a similar fashion to the RF-BD (Section (4.5.2)), the vast majority of false-positive classifications were in close proximity to the boundary ground truth, with the mean Euclidean distance being 4.1 voxels. This is illustrated in figure 4.8 which shows the false-positive count versus the distance from their nearest neighbour boundary ground truth voxel. Given that this region is diffused and that shape constraints are employed within the framework to help segmentation, this boundary detector was sufficient.

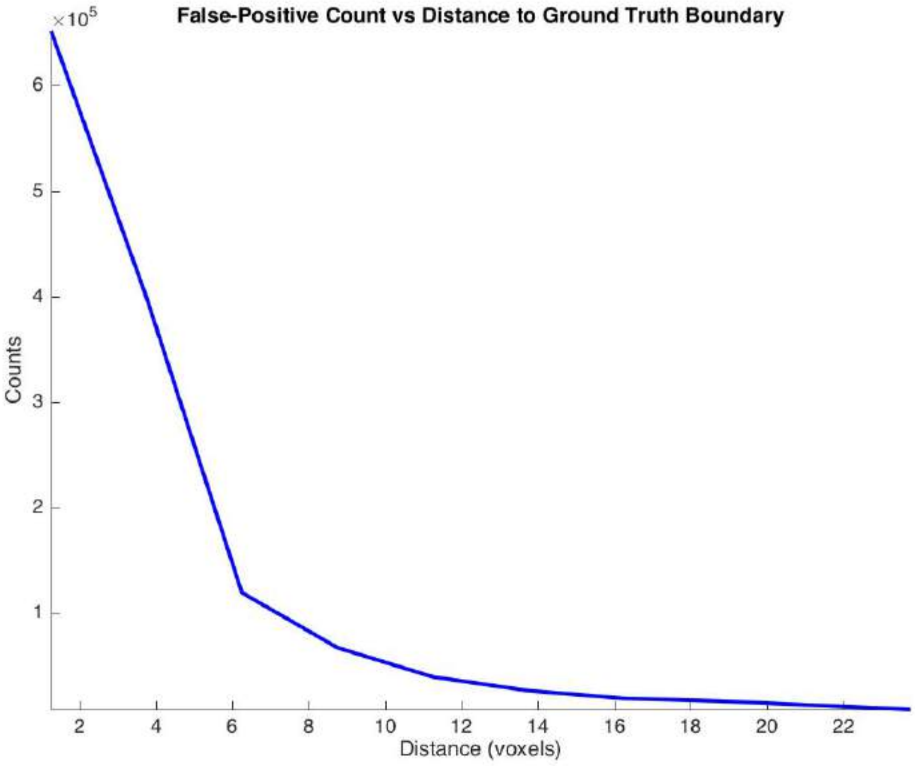


Figure 4.8: Histogram of nearest neighbour distances with NN-BD.

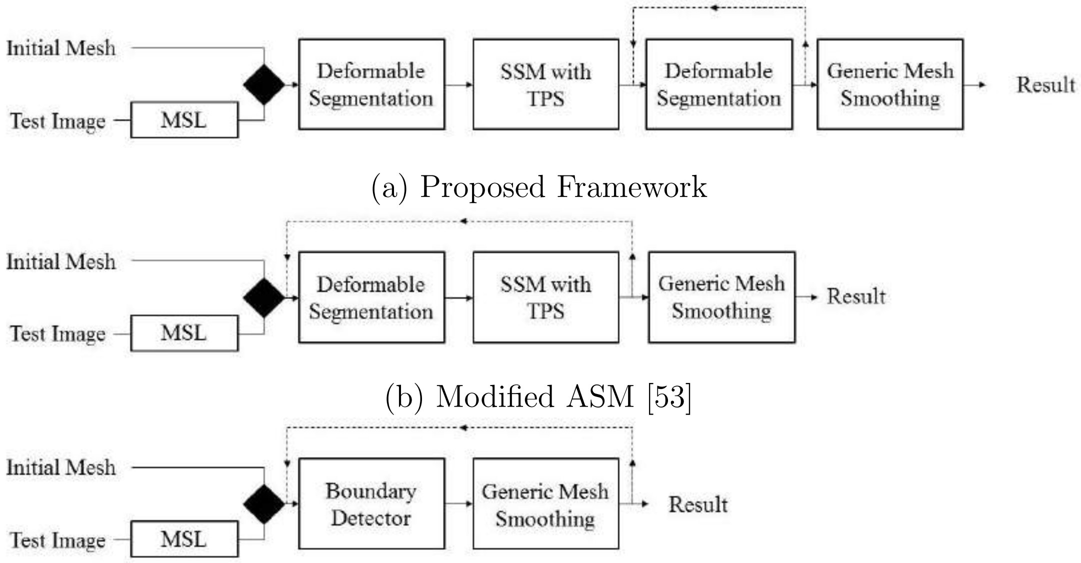
4.6.3 Automatic Correspondence for SSM

To significantly reduce manual work and training time, point correspondences are automatically found across the training set using the technique presented in Chapter 3. Rather than manually labelling correspondences on all training meshes, strategically placed *fiducial* points are labelled on the surface of a single target mesh from the training set, which are then propagated across the remaining meshes in the set.

A target mesh $M_t = (V_t, E_t, F_t)$ with $|V_t| = n$ vertices is randomly selected from the training set, and a subset of $m = 70$ vertices, called *fiducial points*, are labelled such that $P_t \subset V_t$, and $m << n$. All other meshes in the set are regarded as source meshes, such that $M_s = (V_s, E_s, F_s)$ where $|V_s| = p$, and $n \neq p$. The aim now is to find a subset of m vertices $P_s \subset V_s$, that are correspondent with P_t.

As is described in Chapter 3, this is based on the assumption that finding correspondences between two shapes becomes much easier if the shapes are similar to each other. Therefore a transformation is applied $T(x, y, z) : M_s \mapsto M_t$, consisting of global $T_g(x, y, z)$ and local $T_l(x, y, z)$ transformations. T_g globally aligns both meshes and is formulated as an affine transformation from ground truth vectors. T_l then takes into account the local differences in shape, and is estimated using equations (3.4), (3.5), (3.6), (3.7) and (3.8). This time the similarity metric $E(V_t, T_l(V_s))$ is measured as between all vertices in both meshes. Then for every point in P_t its nearest neighbour based on Euclidean distance is found in V_s', resulting in P_s'. Finally, applying T_l^{-1} to P_s' yields P_s.

4.7 Comparative Analysis and Results



(a) Proposed Framework

(b) Modified ASM [53]

(c) Zheng et al. [285] Method

Figure 4.9: Comparison study pipelines.

For comparison, two additional approaches were implemented and tested on the same dataset. Firstly a modified ASM approach was implemented. This enabled the comparison between the proposed framework, which only applied shape constraints once after initial segmentation, and ASM which applied constraints iteratively. In addition, a state-of-the-art deformable modelling approach [285] was implemented, which allowed comparison between introducing some mesh regularisation, and an approach with no regularisation at all. Schematic diagrams of the proposed framework alongside the comparison studies are shown in Figure 4.9.

ASM with a learning-based boundary detector would conventionally be implemented with an iterative process of boundary detection followed by shape constraint [53]. However applying ASM in a conventional manner for this dataset led to severe entanglement due to the complexity of the meshes and image noise. As a result, for fair comparison it was decided to include the B-spline mesh regularisation and TPS warping in a modified implementation of ASM. This essentially became an iterative process of the deformable segmentation component as presented in Section (4.4.2), and SSM with TPS as presented in Section (4.4.3). Generic mesh smoothing [248] was also applied after the iterative process. In addition, MSL was implemented as presented in Section (4.5.1).

The method proposed by Zheng et al. [285] was recreated as close as possible to what was outlined in the original paper, however for fair comparison, MSL was again implemented as presented in Section (4.5.1). The deformable modelling stage was an iterative process consisting of a learning-based boundary detector and generic mesh smoothing [248]. The boundary detector was trained with steerable features and an AdaBoost-based algorithm as was implemented in the original paper.

All experiments were conducted on 36 aortic root volumes in a threefold cross-validation fashion. For every training image, the aortic root, including the ascending aorta and aortic arch were manually labelled in CT TAVI images, of size $512 \times 512 \times (500 \sim 800)$, and the voxel size was $0.48mm \times 0.48mm \times 0.62mm$. In order to ensure that the valve hinges were clearly seen, multi-planar rotation software was used during image labelling to try and get the best view. Meshes were generated from the labels, and the 10 anatomical landmarks were manually labelled. The ascending aorta and aortic arch were then discarded from the mesh below the sinotubular junction to leave a mesh of the aortic root, and a ground truth vector including the root's position, orientation and scale were determined. The average local mesh size was $35mm \times 34mm \times 24mm$.

The segmentation accuracy was quantitatively evaluated using a symmetric point-to-mesh $\overline{PMD}$ distance error. This was calculated as the mean of the Euclidean distances between the nearest neighbours of the ground truth mesh vertices and the resulting mesh vertices, and vice-versa to make the measurement symmetric. This is the most popular evaluation metric used in the literature [64, 72, 100, 262, 288]. The symmetrical Hausdorff distance $\overline{HD}$ was also used to give an idea of any outlying regions in the resulting mesh compared to the ground truth. This was a measurement of the maximum Euclidean distance between nearest neighbours. In a few cases, a small percentage of vertices were stretched below the plane of the sinotubular junction after mesh deformation. In order to ensure that aortic root segmentation was quantitatively evaluated and not parts of the ascending aorta, these vertices were not considered.

In terms of efficiency, both implementations performed the segmentation in ~ 5 seconds, which is comparable to that of Zheng et al. [288], with a time of ~ 4 seconds. Given that the code was not yet optimised, it is envisaged that the testing speed can decrease, although even in its current state it has been shown that the framework is comparable to the state-of-the-art for testing efficiency. It is difficult to accurately quantify the acceleration in training time between the RF and NN implementations of the proposed framework, however it is estimated that the NN implementation was dramatically quicker. The time for the RF-MSL and RF-BD classifiers to train was ~ 3 hours, whereas to train both NN-MSL and NN-BD took ~ 8. However, no time was taken to hand-craft features for the NN implementation, which eliminated lengthy trial and error processes that were necessary for RF implementation. Furthermore, the process of manual correspondence labelling for the RF implementation lasted ~ 2 days, dramatically increasing the model preparation time. The automatic correspondence identification for NN implementation however took $\sim$ 90 seconds for the entire training set, which is a significant time drop.

Figure 4.10 shows an example of the segmentation results of a single example for both implementations of the proposed framework. Figures 4.11, 4.12, 4.13 and 4.14 show qualitative segmentation comparisons on cropped image slices for illustration, where columns are the test slices and rows are the results from the different methodologies. Finally, Figure 4.15 shows the corresponding resulting meshes.

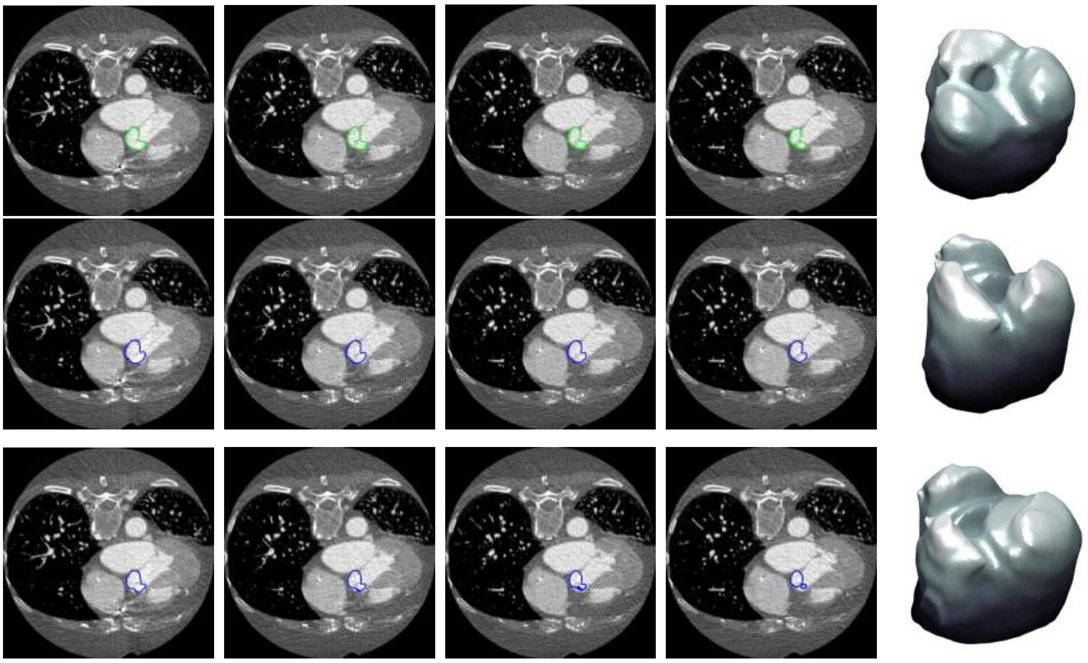


Figure 4.10: Example segmentation results using proposed framework. from top to bottom: 1$^{\mathrm{st}}$ row: Ground truth; 2$^{\mathrm{nd}}$ row: Segmentation result with RF implementation; 3$^{\mathrm{rd}}$ row: Segmentation result with NN implementation.

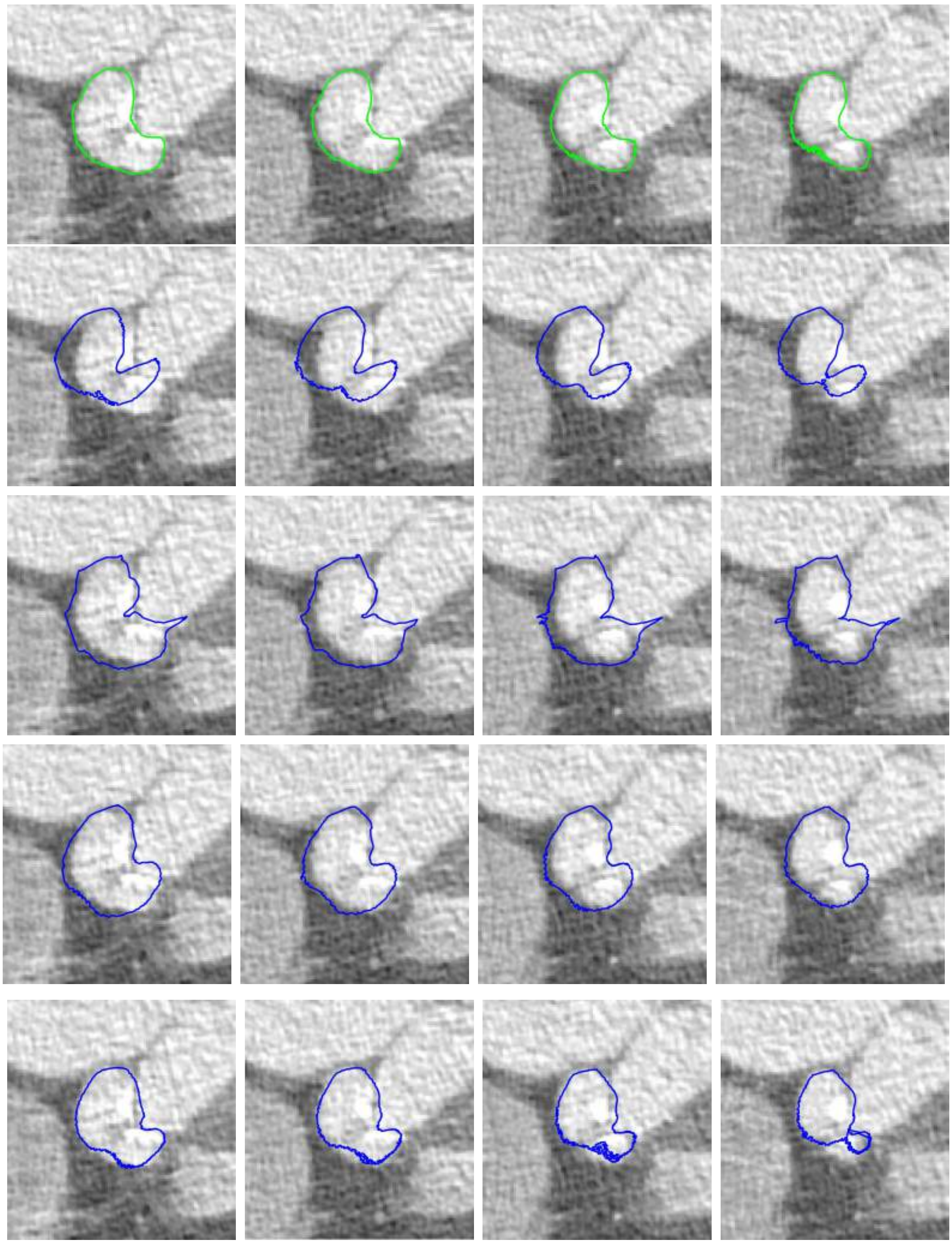


Figure 4.11: Example 1: Segmentation comparisons. From top to bottom: 1st row: Ground truth; 2nd row: Modified ASM; 3rd row: Zheng et al. [285] method; 4th row: Proposed framework with RF implementation; 5th row: Proposed framework with NN implementation.

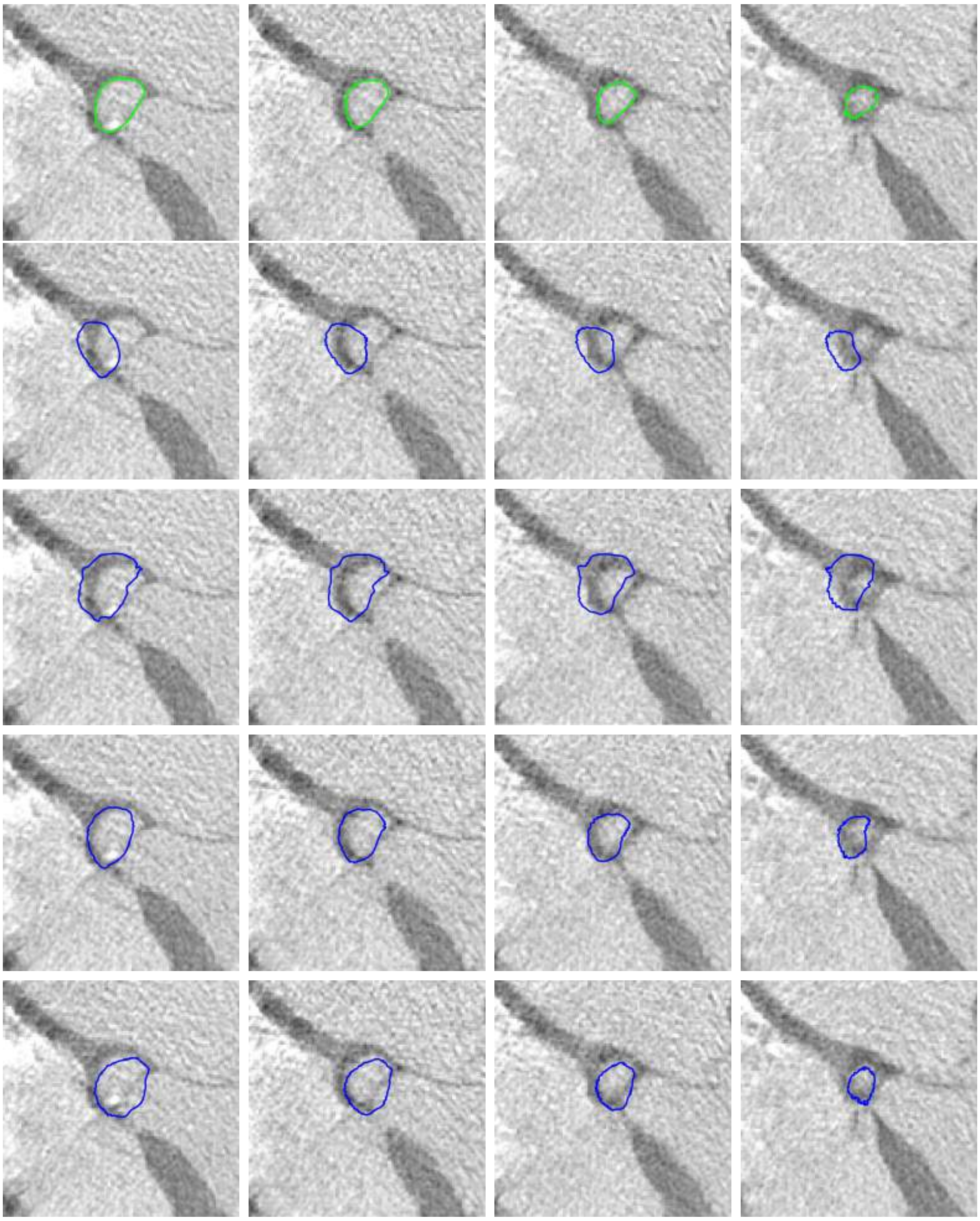


Figure 4.12: Example 2: Segmentation comparisons. From top to bottom: 1st row: Ground truth; 2nd row: Modified ASM; 3rd row: Zheng et al. [285] method; 4th row: Proposed framework with RF implementation; 5th row: Proposed framework with NN implementation.

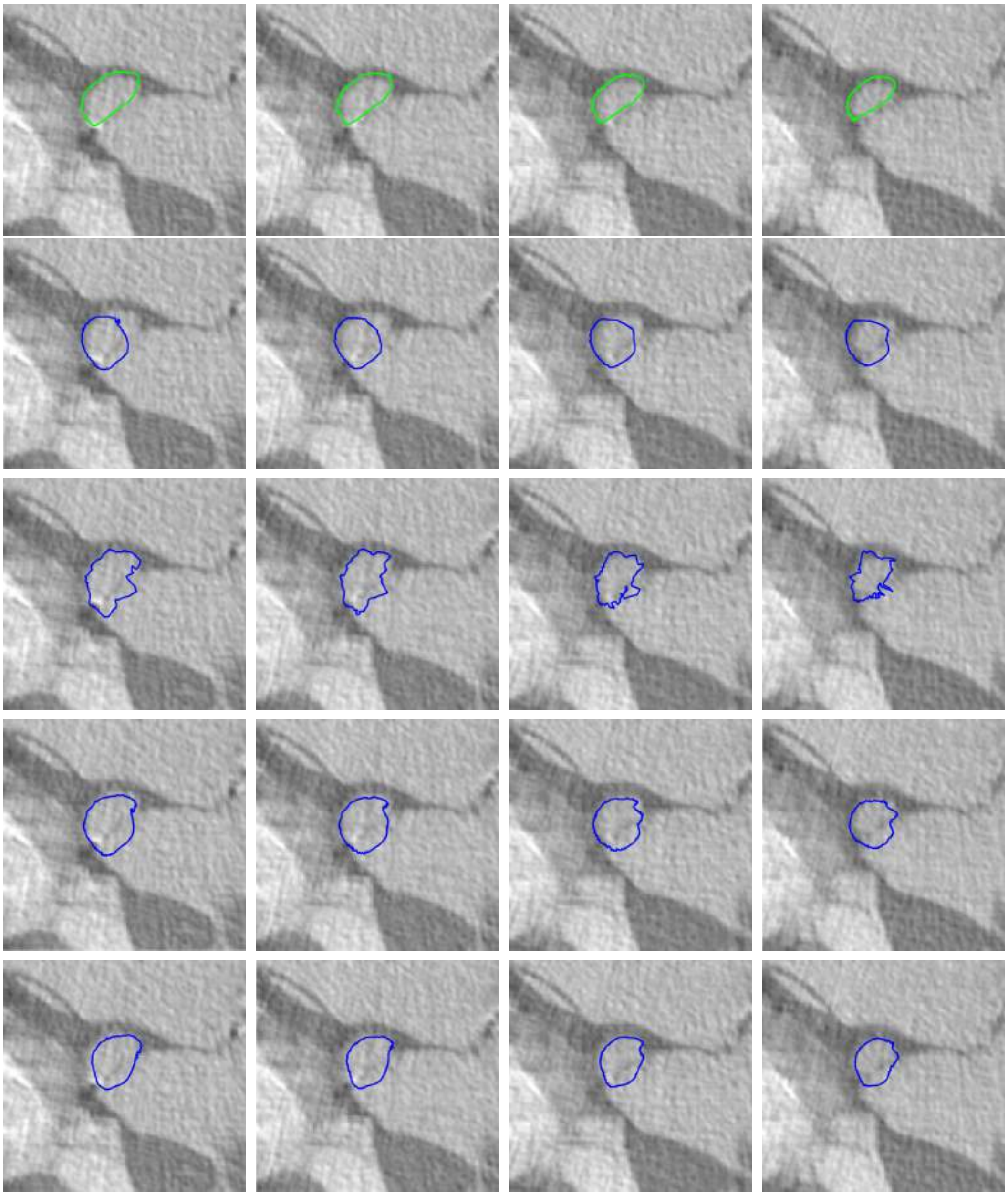


Figure 4.13: Example 3: Segmentation comparisons. From top to bottom: 1st row: Ground truth; 2nd row: Modified ASM; 3rd row: Zheng et al. [285] method; 4th row: Proposed framework with RF implementation; 5th row: Proposed framework with NN implementation

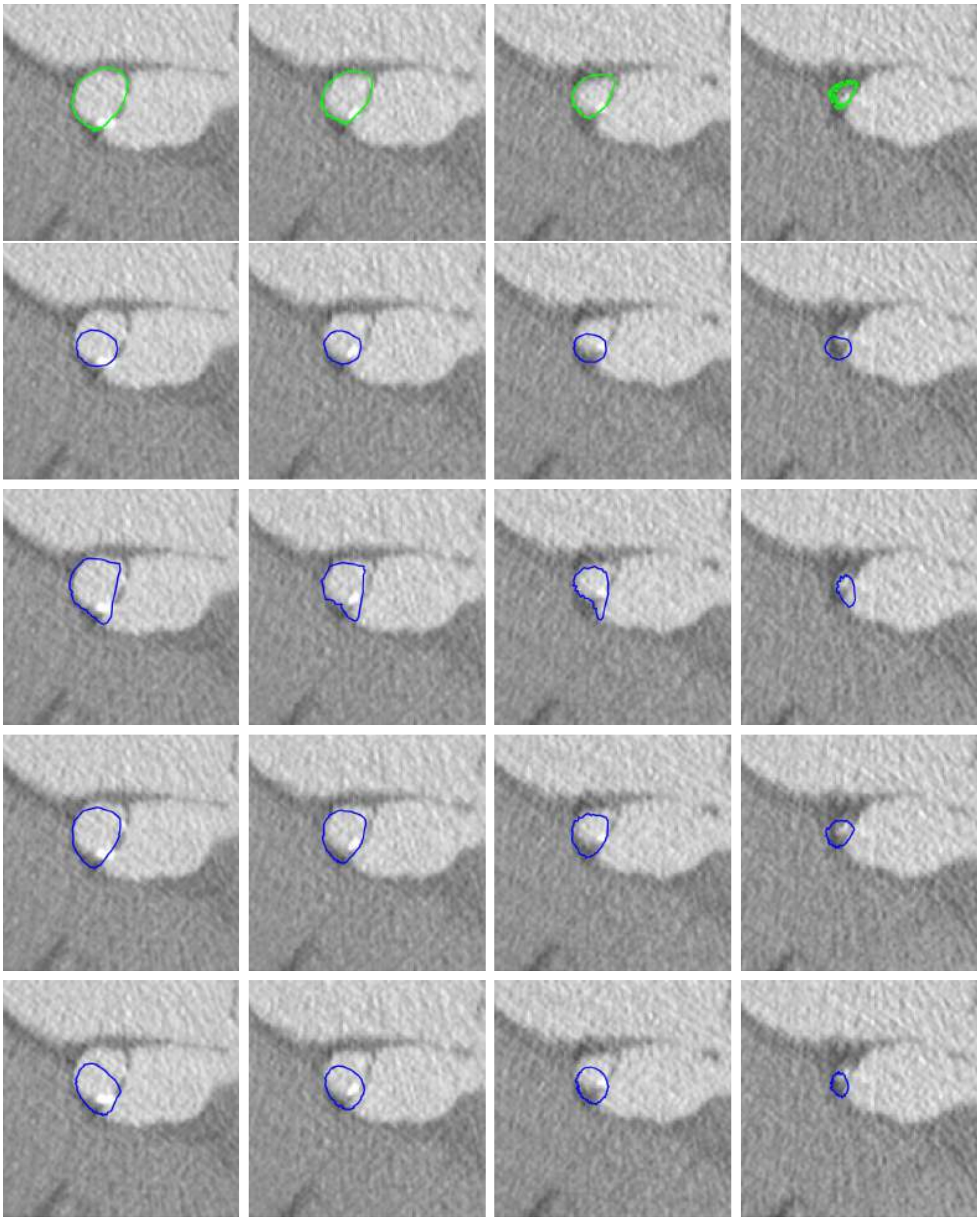


Figure 4.14: Example 4: Segmentation comparisons. From top to bottom: 1st row) Ground truth; 2nd row: Modified ASM; 3rd row: Zheng et al. [285] method; 4th row: Proposed framework with RF implementation; 5th row: Proposed framework with NN implementation

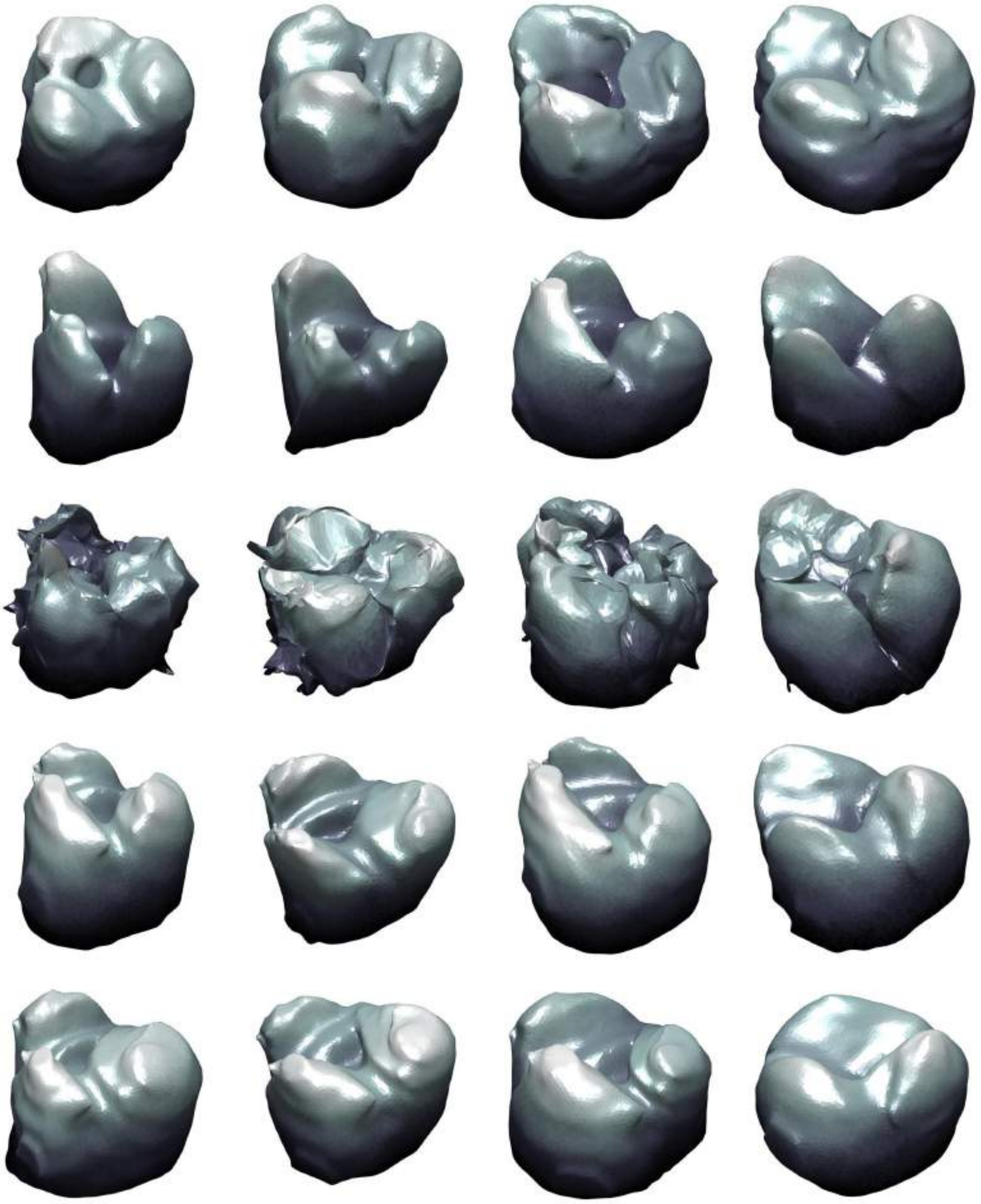


Figure 4.15: Corresponding comparison mesh results. From left to right: 1st column: Mesh results for Figure 4.11; 2nd column: Mesh results for Figure 4.12; 3rd column: Mesh results for Figure 4.13; 4th column: Mesh results for Figure 4.14. From top to bottom: 1st row: Ground truth; 2nd row: Modified ASM; 3rd row: Zheng et al. [285] method; 4th row: Proposed framework with RF implementation; 5th row: Proposed framework with NN implementation

Method	$\overline{PMD}$ **(mm)**	$\overline{HD}$ **(mm)**
Modified ASM	1.86 ± 0.46	8.87 ± 2.79
Zheng et al. [288]	1.47 ± 0.51	10.79 ± 2.66
Proposed Framework with RF Implementation	$\mathbf{1.39 \pm 0.29}$	$\mathbf{6.75 \pm 2.05}$
Proposed Framework with NN Implementation	1.61 ± 0.33	8.41 ± 2.80

Table 4.1: Quantitative comparison.

Table 4.1 presents and compares the quantitative results of the proposed framework to that of [285, 288], and the modified ASM method.

The proposed framework produced highly comparable results to manual segmentations. With RF implementation it obtained an average $\overline{PMD}$ of $1.39 \pm 0.29mm$, which translates to $\sim$ 3 voxels, and the average $\overline{HD}$ was $6.75 \pm 2.05mm$. Similarly, NN implementation yielded a $\overline{PMD}$ of $1.61 \pm 0.33mm$ and $\overline{HD}$ of $8.41 \pm 2.80mm$.

The proposed framework with RF implementation produced the lowest point-to-mesh error with a distance of 1.39mm. The framework with NNs as well as the method by Zheng et al [285] also produced comparable point-to-mesh errors, which are all lower than that produced by the modified ASM. This is reflected in Figures 4.11, 4.12, 4.13 and 4.14. This approach produced a significantly higher mesh error which suggests that applying strong shape constraints on every boundary refinement iteration was too restrictive. As a result the method did not allow enough freedom for the mesh to deform towards the boundary. The failure of this method is also due to the fact that vertex-level correspondence in smooth and complex anatomical structures is very difficult to obtain and local deformation can be non-linear.

Figure 4.15 highlights the difference in mesh appearance between all four methods. Both implementations of the proposed framework as well as the modified ASM employ B-spline mesh regularisation, which has a dramatic effect on the mesh surfaces. These resulting meshes have no compressed or stretched mesh faces and no tangled mesh edges, leaving a smooth regularised mesh surface.

The iterative approach proposed in [285] produced meshes that were severely disordered, which was due to the absence of any mesh regularisation after vertex boundary detection. The only form of regularisation here was some generic mesh smoothing [248] which was applied after boundary detection. This was suitable to smooth faceted mesh faces, but could not cope with untangling crossed mesh edges. In reality 15 iterations of this boundary refinement approach were attempted, however in all 36 cases the mesh became tangled after three or less iterations. The results presented here for this method are the best ones.

The mesh appearance is also reflected in the Hausdorff distance results. The proposed framework with RF implementation produced the lowest error with a distance of 6.75mm which was down to effective mesh regularisation. Meanwhile the method by [285] produced an average Hausdorff distance of 10.79mm, which was notably higher than that of the other three methods. This suggests that large regions of the resulting meshes deviated significantly from the ground truth, with the most likely reason being the lack of any mesh or shape regularisation.

Overall, it is clear that the proposed framework with RF implementation produced the best segmentation results, with significantly lower quantitative errors and highly regularised mesh appearance. Comparing the Zheng et al. [285] results with that of the modified ASM shows that mesh regularisation is necessary, however too much regularisation restricts sufficient mesh movement. The proposed framework yields the best results by achieving balance between shape regularisation to allow smooth mesh surfaces, but allowing enough freedom for the mesh to deform towards the object boundary in the image. This is achieved by only applying shape constraints once after initial segmentation, which avoids overly relying on strong prior shape constraints, and makes the framework more image-driven.

While the proposed framework with RF implementation produced both the lowest point-to-mesh and Hausdorff distances, applying this framework with NN implementation also produced comparable results, yielding similar smooth mesh surfaces and the second lowest Hausdorff distance error. This indicates that the proposed framework itself is strongly responsible for the smooth mesh appearance due to the balance between strong shape constraints and 3D mesh regularisation. Furthermore, this also suggests that there was little difference between the effects of the manually generated SSM and automatically generated SSM, indicating that the automatic method (Chapter 3) is sufficient for automatically obtaining point correspondences.

The point-to-mesh distance produced with NN implementation is higher than that produced with RF implementation, and although it is also marginally higher than that of [285], the results are much more regularised. Table 4.2 clearly shows the reason for this, as the RF-MSL pose estimation is better than that of NN-MSL. Although the orientation error is very similar, the RF-MSL position estimator produced more than half the error of NN-MSL. This shows that the Haar and SHOG features were more effective for aortic root detection than the *learned* features produced by the NN. As a result this had a knock-on effect on the segmentation results. However, given that the position search space was $512 \times 512 \times (500 \sim 800)$, an Euclidean distance error of 12 voxels can be considered sufficient, although improvements can be made to produce performances similar RF-MSL. On the other hand, Table 4.2 also shows that the NN-BD had higher sensitivity in detecting the boundary than the hand-crafted RF-BD. This suggests that the *learned* features from the NN had more discriminative power than the Haar feature extracted for the RF detector.

In addition, a further experiment was conducted by implementing a semi-automatic version of the proposed framework. This consisted of using ground truth pose-alignment rather than MSL, followed by the proposed deformable modelling stage with RF-BD features. This produced a point-to-mesh error of $1.19 \pm 0.21mm$

Implementation	RF	NN
Position Error (voxels)	5.6 ± 2.9	12.2 ± 4.2
Orientation Error (degrees)	13.8 ± 10.9	15.4 ± 11.0
BD Sensitivity (%)	54.5	66.2
BD Specificity (%)	88.4	86.5
$\overline{PMD}$ (mm)	1.39 ± 0.29	1.61 ± 0.33
$\overline{HD}$ (mm)	6.75 ± 2.05	8.41 ± 2.80

Table 4.2: Comparison results of test images between both implementations of the proposed framework.

and a Hausdorff distance of $5.79 \pm 1.76mm$, suggesting that improvement to the pose estimation can lead to even better segmentation. Furthermore, calcification around the aortic root often appears in cardiac images which makes boundary detection difficult as the true boundary cannot be seen. It is thought that this is the cause of the relatively high Hausdorff distance in all results, and segmentation would improve if this problem was addressed. Furthermore, as it was shown that learning-based algorithms were effective, it is thought that by carefully selecting the NN architectures even better segmentation results can be achieved in future work.

The following list summarises the reasons for the differences in segmentation performance between the four implementations.

Modified ASM [53]:

- High $\overline{PMD}$ & poor segmentation - Too much shape regularisation and not enough flexibility for model to reach areas of high curvature.
- Lower $\overline{HD}$ & smooth mesh appearance - Strong shape regularisation ensuring regularised surface.

Zheng et al. [288]:

- Low $\overline{PMD}$ & good segmentation - Not a lot of shape regularisation, and plenty of flexibility for model to reach areas of high curvature.

- High $\overline{HD}$ & irregular mesh appearance - Not enough shape regularisation to ensure regularised surface.

Proposed Framework with RF Implementation:

- Low $\overline{PMD}$ & good segmentation - Not too much shape regularisation, and plenty of flexibility for model to reach areas of high curvature.
- Low $\overline{HD}$ & smooth mesh appearance - Enough shape regularisation ensuring regularised surface.

Proposed Framework with NN Implementation:

- Low $\overline{PMD}$ & good segmentation - Not too much shape regularisation, and plenty of flexibility for model to reach areas of high curvature.
- Low $\overline{HD}$ & smooth mesh appearance - Enough shape regularisation ensuring regularised surface.
- Higher $\overline{PMD}$ & $\overline{HD}$ than RF implementation - Accuracy errors in MSL pose estimation having a knock on effect on final segmentation.

4.8 Summary

In this chapter, a fully-automatic deformable modelling-based framework for the 3D segmentation of the aortic root in cardiac CT images was presented. Two implementations of the framework were carried out, one with a hand-crafted RF implementation and one with a non-hand-crafted NN implementation. The first implementation employed RF-MSL classifiers and a boundary detector that were trained with manually selected features which were carefully chosen to achieve the most discriminative power. In addition, the SSM was implemented using manually labelled mesh correspondences. The second implementation significantly reduced the training time by automatically obtaining correspondences for the SSM, and by using NNs which learned useful features rather than carefully hand-picking them.

The proposed framework was very efficient in pose estimation and deformable segmentation, and allowed enough shape regularisation to maintain a regularised mesh surface, while also allowing enough freedom for sufficient deformation. This intrinsic mesh regularisation allowed large deformations to take place. It was also shown that statistical shape regularisation was effective at the coarser scale and non-linear local refinement was necessary to archive accurate segmentation. In addition, the hand-crafted object and boundary detection features were efficient and robust in defining aortic roots, and the learned features showed promise for deformable modelling. Both qualitative and quantitative results showed that the proposed framework outperformed the state of the art, as well as the classical ASM. This justified the decision to build an image-driven segmentation method rather than a top-down approach which emphasises on shape priors.

Chapter 5

Lymphatic Vessel Segmentation

Contents

5.1 Introduction

This chapter presents a further study for deformable modelling on 3D ex-vivo confocal microscopy images of the lymphatic vessel. Deformable models have been used specifically for vessel segmentation [70,74,117,136,151,280], by using bottom-up image-driven constraints as well as some top-down prior shape knowledge. As such they have the ability to overcome appearance inconsistencies which are often present in such images. Successful implementations include work by Ecabert et al. [70], who segmented large heart vessels with a 3D mesh model. A combination of shape priors from a SSM and gradient-based boundary detection was used, and the model's flexibility increased with every iteration to allow more degrees of freedom. A similar approach was adopted by Yim et al. [280], who performed angiographic vessel segmentation by deforming a 3D mesh. Alternatively, Hu et al. [117] segmented these vessels on 2D MRI images, and was accomplished by formulating an energy function to deform snakes towards the cross-sectional boundary. In a different application Espona and Ortega [74] segmented the retinal vascular tree in 2D images using a simplistic snake model. Much emphasis was put on vessel detection in this work, as retinal images are noisy, and contain many tubular-like structures.

Although successful, these approaches have all been applied to well defined vessels in their respected images. Blood contrasts are often used to enhance the appearance of blood vessels in medical images [99], including CT [70], MRI [117] and angiography images [280]. Furthermore, retinal vessels have very good contrast in retinal vascular imaging [74], making boundary detection a simpler task. This chapter explores the use of image-driven deformable modelling for segmenting the lymphatic vessel in highly obscured confocal microscopy images. The challenges with this dataset are;

- As these images are acquired ex-vivo there is no liquid contrast present in the images, making vessel contrast solely a result of tissue type and thickness.
- Due to the nature of confocal microscopy image acquisition, individual cross-sectional vessel slices appear to have strong inner and outer wall appearance to the left of the slice, but is dramatically more obscured to the right. This

makes finding suitable boundary features capable of identifying both high and low contrast wall regions challenging, especially as the appearance of the walls are also diffused.

- The vessels have valves at the centre of the inner wall, which have very similar contrasts to the vessel walls themselves. Making sure that the deforming model does not converge at these valves is also a non-trivial challenge.

Examples of these images is shown in Figure 5.1.

Many existing vessel segmentation approaches work on 2D slices, due to the simple circular shape of a vessel's cross-section [74, 117]. It is also assumed that segmenting multiple 2D circular-like shapes is a simpler challenge than segmenting a 3D tubular-like shape. As such, often for 3D vessel segmentation 2D cross-sectional segmentation is performed first, before being concatenated to produce a final tube-like structure [136, 151]. Such an approach is also adopted in this chapter, where cross-sectional circular-like segmentations are performed and concatenated to generate a 3D mesh, before treating the 3D mesh as a whole for surface regularisation.

The aim is to use a similar bottom-up framework used in the previous chapter on a new challenging dataset, where edges are highly obscured and inconsistent between image slices. The deformable model approach presented here combines a learning-based boundary detector with mesh regularisation, where it is shown that neural networks are capable of identifying boundaries of highly varying appear-

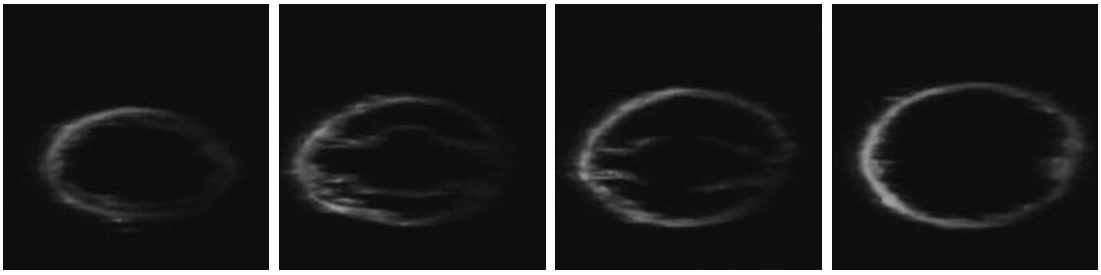


Figure 5.1: Series of 2D confocal microscopy slices highlighting the contrast challenges associated with the segmentation.

ance. This not only allows potentially abstract features to be learned, but also eliminates manual feature preparation and accelerates training. Shape constraints from a SSM are not required due to the nature of the vessel's tubular shape, and it is suggested that even the softest shape constraints on such a shape may be detrimental by being too restrictive still. As a result the shape regularisation comes solely from the B-spline regularisation component.

As such the proposed framework is compared against a simple intensity-based approach without any shape regularisation at all, and another intensity-based approach with too much regularisation. Furthermore, the framework is also compared to a graph-cut approach with a Hidden Markov model (HMM) [75], to show that image-driven deformable models with little shape regularisation can compete with other shape prior-based methodologies.

Boundary detectors in deformable modelling are often used by defining search paths along the surface normal direction, in order to try and avoid surface entanglement. The search path coordinates with the strongest boundary responses are then taken as the contour points' new position. Classification algorithms are necessary to drive the initial model towards the object boundary, and have been used in medical image deformable modelling on numerous occasions [138,166,173,288].

Often, these studies consist of hand-picking useful features to distinguish between boundary and non-boundary pixels. Steerable features [83,288], Haar features [261,285], Canny edge features [37,138] and gradient steerable features [166] have all been used. However, choosing appropriate features to use is not a trivial task as useful features are very much data-dependent, with object type, image modality and image contrast all effecting the usefulness of a feature. In addition, highly abstract features may be very useful for boundary detection, therefore by using hand-crafted features it is possible to miss out on some additional, potentially useful information. From the previous chapter it has been shown that NNs are capable of learning features by adjusting the weight of each node using backpropagation, therefore they are popular learning systems for image recognition, and have been used specifically for edge detection [164,227].

5.2 Contribution & Overview

An investigation is carried out for the use of multiple neural network architectures as learning-based boundary detectors for deformable model-based segmentation. The experiments are performed on 3D confocal microscopy images of the lymphatic vessel, however a 2D NN is trained here for slice-by-slice classification. With such inconsistent vessel wall appearance, this dataset makes choosing features capable of identifying all boundary pixels very challenging indeed. Furthermore this phenomenon occurs at both the inner and outer vessel walls. As a result, two classification processes are performed; is the pixel on the vessel's outer wall or not; or is it on the inner wall not? Raw pixel intensities from a local patch, rather than hand-crafted features are extracted, and used as inputs so that the network may learn useful, potentially abstract features which can deal with such boundary inconsistencies.

After boundary detector testing the appropriate detector is used for full segmentation. The framework consists of a simple and efficient intensity-based initial segmentation, followed by the learning-based boundary detector and 3D mesh regularisation in an iterative fashion.

5.3 Image-Driven Segmentation Framework

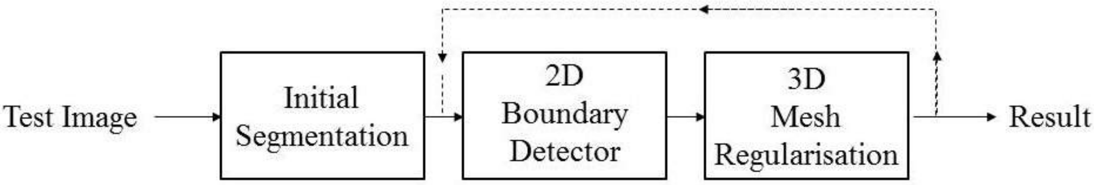


Figure 5.2: Overview of the proposed lymphatic vessel segmentation at the testing stage.

The proposed deformable modelling framework consists of an initial segmentation based on a simple intensity filter on each individual image slice, which is used to generate an initial mesh model. Following this, an iterative deformable modelling process is implemented which deforms the initial mesh towards the lymphatic vessel wall. An overview of the proposed framework is shown in Figure 5.2. For full vessel segmentation, the segmentation framework is implemented twice; once for the outer wall and once for the inner wall; so that segmentation of both walls are carried out independently.

The initial mesh is generated by first carrying out an initial segmentation on each individual image slice. These segmentation contours have the same number of points, making it possible to define edges between slices to generate a face-vertex mesh.

The deformable modelling process is iterative and stops when the maximum number of iterations is reached. This process consists of two components. Firstly a 2D boundary detector is carried out on each individual slice, where search paths are defined along the normal direction of each mesh vertex. A neural network is used for boundary detection in order to learn useful features, rather than spending time hand crafting them. Finally, 3D mesh regularisation is implemented on the entire mesh, using the B-spline-based method used in Chapter 4. This ensures a smooth surface not only on each 2D contour, but also between contours in the third dimension.

On every iteration the boundary detector's search path decreases in order to reach convergence, and the degrees of freedom associated with the mesh regularisation increases to allow the mesh to deform to areas of high curvature. As a result, the deformable modelling process can be thought of as an iterative refinement process.

The remainder of this section is as follows. Section (5.3.1) describes the initial segmentation process which also generates the initial mesh model. Section (5.3.2) describes the boundary detector, with experiments to determine which NN architecture to implement. Finally, Section (5.3.3) describes the B-spline-based 3D mesh regularisation.

5.3.1 Initial Segmentation

Before deformable modelling, an initial mesh model must be defined. This section describes a quick initial 2D intensity-based segmentation on a slice-by-slice basis, before combining the estimated 2D contours to generate a simple 3D mesh structure.

It is assumed that the pixel intensities within the lymphatic vessel (i.e. between the inside and outside walls) are significantly higher than the remaining pixels in the image slice. Therefore it is assumed that the pixels with highest local gradient are on/close to the boundary. To this end, a simple 2-rectangle Haar-like filter [261] is employed to highlight the pixels of high gradient. For every pixel in the image a filter response is computed as follows;

$$f_{out} = \sum_{k=0}^{N} \mu_1^k - \sum_{k=0}^{N} \mu_2^k \tag{5.1}$$

where μ_1^k is the intensity of pixel k in rectangle 1, μ_2^k is the intensity of pixel k in rectangle 2, and N is the number of pixels in each rectangle.

Given that an optimal Haar filter response would have high pixel intensities in one rectangle and low pixel intensities in another, it is necessary for the filter to be orthogonal to the boundary itself. As we do not yet know the location and direction of the tubular-like boundary in cartesian coordinates, the filter is applied to the image in polar coordinates. An example is shown in the left column of Figure 5.4, where for each column in the polar image the pixel with the optimal filter response is identified. As the appearance of the outer and inner walls are opposite each other, so too will the filter response. Therefore the maximum value of Equation 5.1 is used to identify pixels on the outer wall, and the maximum value of;

$$f_{in} = \sum_{k=0}^{N} \mu_2^k - \sum_{k=0}^{N} \mu_1^k \tag{5.2}$$

is used to identify the pixels on the inner wall. A filter size of 1 × 21 is centred at the test pixel, where $N = 10$ pixels are summed in rectangle 1, and $N = 10$ pixels are also summed in rectangle 2.

The contour in polar coordinates is then converted back to cartesian coordinates, and is smoothed by fitting the contour to an ellipse. This is done by optimising the conic equation for an ellipse;

$$ax^2 + bxy + cy^2 + dx + ey + f = 0 \tag{5.3}$$

where a, b, c, d, e and f are parameters to be estimated, and x and y are the 2D contour point coordinates. Optimisation is achieved using the least-squares algorithm [43]. Example smoothed contours in cartesian coordinates are also shown in Figure 5.4.

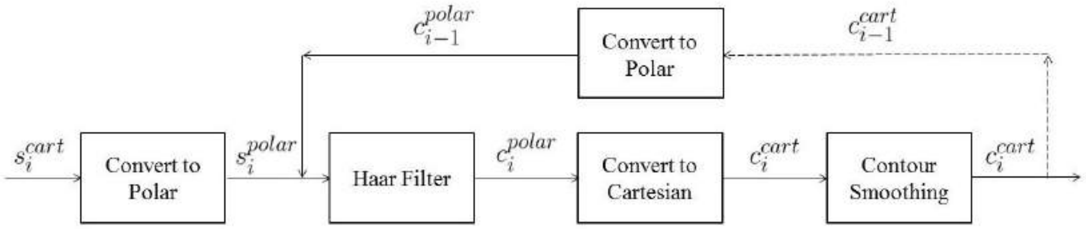


Figure 5.3: Overview of the proposed initial segmentation.

This process is repeated for every slice in the 3D image. However, it is also assumed that the boundary walls do not significantly change between adjacent slices, therefore the smoothed contour from the previous slice is used to help estimate the contour on the next slice. This is done by restricting the search space for finding the optimal filter response in each polar image column. A contour point in column j in the polar image of slice $i + 1$ must be within $\pm l$ of the contour point in column j in slice i.

Figure 5.4 shows an illustration of this. The Figures in the left column show example filter results in polar coordinates for slice i, and the middle column shows the corresponding smoothed contours in cartesian coordinates. The Figure in the right column now shows the contour estimation for the next slice $i+1$. The yellow contours represent the polar image search space limits. These are computed by converting the smoothed contour of slice i back into polar coordinates, and adding leeways of $\pm l$. Now the optimal filter response is only searched for between these two limits in each column, resulting in the contour estimation (blue).

Using these search space limits essentially introduces soft shape constraints to the initial segmentation, and ensures continuity between segmentations on adjacent slices. Figure 5.3 illustrates the initial segmentation process for all slices, where s_i^{cart} and s_i^{polar} are the images for slice i in cartesian and polar coordinates, and c_i^{cart} and c_i^{polar} are the estimated contours for slice i.

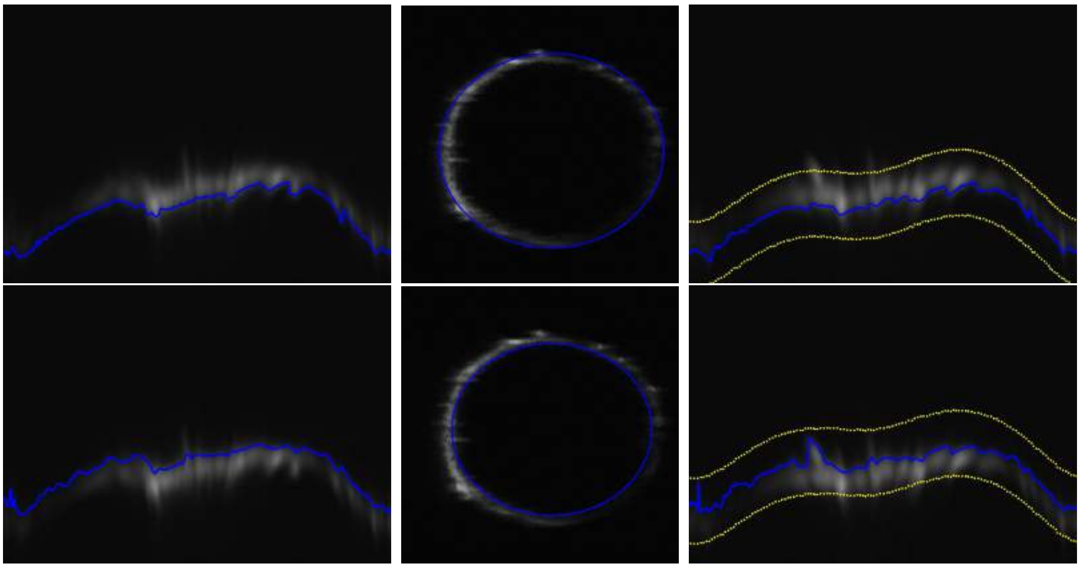


Figure 5.4: Top row: Outer wall; Bottom row: Inner wall. From left to right; 1st column: 2D slice in polar coordinates s_i^{polar}, with outer wall contour after filtering c_i^{polar}. 2nd column: smoothed outer wall contour in cartesian coordinates c_i^{cart}. 3rd column: s_i^{polar} of next slice, with inner wall contour c_i^{polar} and spatial limits (yellow).

Given that each slice was converted to polar images of the same size, the number of contour points on each slice is also equal. Therefore contour point j in slice i is correspondent to contour point j in slice $i+1$. This makes generating the mesh a simple task by simply defining mesh edges between corresponding contour points in adjacent slices.

5.3.2 Boundary Detection

Multiple NN architectures trained with backpropagation were tested for classifying the outer and inner lymphatic vessel walls on 2D image slices, before deciding on a suitable architecture to embed in the segmentation framework. To simulate a conventional deformable modelling process, the ground truth contour is aligned at the centre of the test image. At either side of the contour points a search path is defined along the normal direction. The normal directions can be straightforwardly computed given the contour point's neighbours. For the purpose of testing the detector, each search path coordinate is tested to get a classification result.

Raw intensity values from a local patch are inputted into the network for each sample. To ensure that the appearance of the boundary pixels' local patches are rotationally invariant, the local patch is aligned with the search path. This comes at no extra computational cost as the normal directions have already been computed in order to define the search paths. A schematic diagram of this process is shown in Figure 5.5.

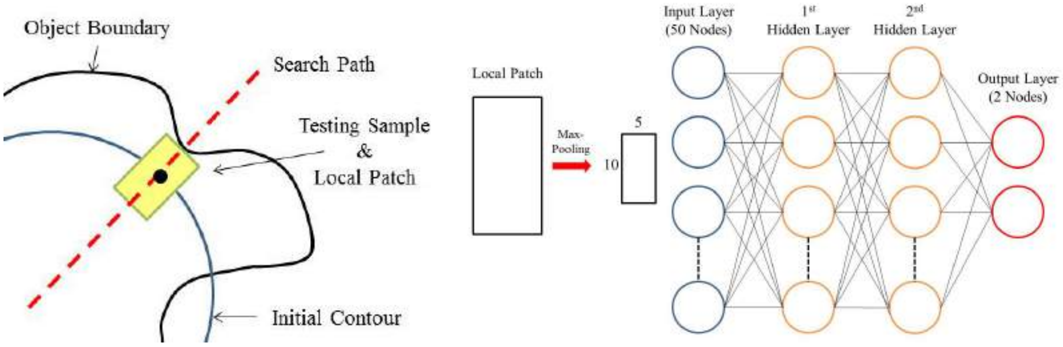


Figure 5.5: Left: Schematic diagram of local patch intensity extraction. Right: Example NN architecture with 2 hidden layers.

Local patches are down-sampled in order to reduce the number of network inputs, which subsequently speeds up the training process. *Max-pooling* is used for this purpose. The process involves sliding a non-overlapping pooling window across the image and extracting the maximum intensity value in each window. While this is an effective down-sampling technique, it also creates position invariance over larger local regions. The size of the pooling window and its stride are chosen to ensure a 10 × 5 output in all cases.

The fully-connected neural network architecture is constructed in the conventional manner, with the 50 pixels of the local patch being represented by an input layer of 50 nodes. The output layer consists of 2 nodes representing boundary and non-boundary, essentially making this a binary classification problem. All nodes in adjacent layers are fully connected. Figure 5.5 shows a schematic example of the neural network architecture with 2 hidden layers.

The lymphatic vessel's outer wall was labelled on six 512 × 512 × 512 *ex-vivo* confocal microscopic volumes. Gaussian smoothing was applied to the images in an attempt to remove noise. In total, three experiments were carried out. Experiment 1 investigated what effect the number of neurons in the first hidden layer had on the classification. Experiment 2 investigated the effect of different local patch sizes, and Experiment 3 investigated the effect of the number of hidden layers in the network. All experiments were run with leave-one-out, where five volumes were used for training in each case.

In all cases fifty thousand sample pixels were used for training, 50% of which were boundary (positive) and 50% non-boundary (negative). Boundary samples were randomly selected from the manually labelled ground truth contours. Search paths were defined for all ground truth contour points, which were aligned along their normal direction with a length of 30 pixels either side of the ground truth pixel. Non-boundary samples were randomly selected from these search paths. At the testing stage a search path of 30 pixels was also defined either side of each initial contour point, resulting in 61 path pixels for each point. All of the search path coordinates at this stage were tested. A summary of the experiments conducted along with their classification results are shown in Tables 5.1, 5.2 and 5.3.

Test	1	2	3	4	5	6
Patch Size	40×20	40×20	40×20	40×20	40×20	40×20
Architecture $[n_1, n_2, \ldots, n_L]$	[1]	[2]	[5]	[15]	[40]	[100]
Inner Sens. (%)	89 ± 7	90 ± 4	91 ± 3	91 ± 3	88 ± 2	90 ± 3
Inner Spec. (%)	74 ± 3	80 ± 4	81 ± 4	83 ± 3	85 ± 3	84 ± 3
Outer Sens. (%)	89 ± 9	88 ± 8	92 ± 5	91 ± 4	91 ± 4	90 ± 5
Outer Spec. (%)	77 ± 4	82 ± 3	84 ± 2	86 ± 2	87 ± 1	86 ± 3

Table 5.1: Experiment 1: Effect of number of nodes in first hidden layer on boundary classification. (n_i is the number of nodes in hidden layer i).

Test	1	2	3
Patch Size	20 × 10	40 × 20	80 × 40
Architecture $[n_1, n_2, \ldots, n_L]$	[40]	[40]	[40]
Inner Sens. (%)	87 ± 5	88 ± 2	90 ± 4
Inner Spec. (%)	83 ± 5	85 ± 3	85 ± 2
Outer Sens. (%)	87 ± 6	91 ± 4	91 ± 3
Outer Spec. (%)	87 ± 2	87 ± 1	86 ± 1

Table 5.2: Experiment 2: Effect of patch size on boundary classification. (n_i is the number of nodes in hidden layer i).

Test	1	2	3
Patch Size	40 × 20	40 × 20	20 × 20
Architecture $[n_1, n_2, \ldots, n_L]$	[40]	[40,20]	[40,20,10]
Inner Sens. (%)	88 ± 2	91 ± 3	92 ± 4
Inner Spec. (%)	85 ± 3	83 ± 3	83 ± 3
Outer Sens. (%)	91 ± 4	92 ± 5	91 ± 5
Outer Spec. (%)	87 ± 1	86 ± 2	86 ± 2

Table 5.3: Experiment 3: Effect of number of hidden layers on boundary classification. (n_i is the number of nodes in hidden layer i).

Inner wall classification sensitivity and specificity from all experiments ranged between 87%-92%, and 74%-85%, respectively, while outer wall sensitivity and specificity ranged between 87%-92% and 77%-87%. It is immediately obvious that the detectors' sensitivity are significantly higher than their specificity. This is to be expected as the boundary region is highly diffused, with regions close to the boundary having significantly higher pixel intensities than those that are further away. Given this, it is important to find an architecture which produces the highest specificity results as possible to accurately classify non-boundary pixels close to the boundary. However, given that this is a first attempt at boundary detection, these results indicate that NNs can produce acceptable results for lymphatic vessel boundary detection.

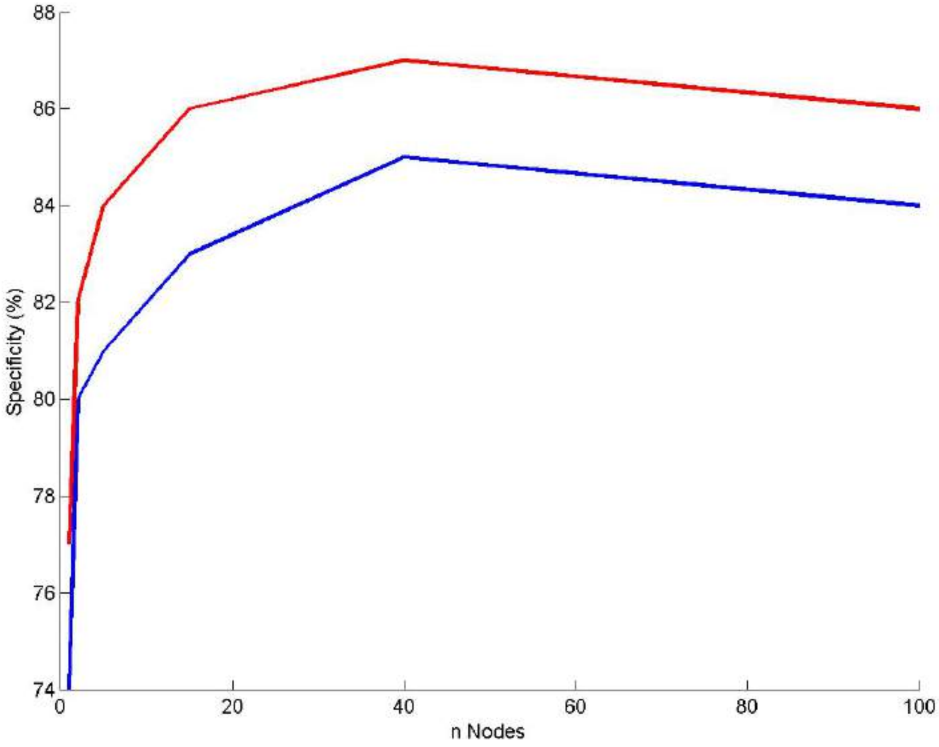


Figure 5.6: Plot of Experiment 1 specificity values. Blue: Inner Wall; Red: Outer Wall.

Experiment 1 highlights the difference that the number of nodes in the network's first hidden layer has on classification. It can be seen that the specificity in both cases increases with an increasing number of nodes until some convergence where the specificity plateaus at 40 nodes. This pattern is shown in Figure 5.6, and suggests that a sufficient number of nodes is necessary in order to discriminate between boundary and non-boundary pixels. The difference after more than 40 nodes is negligible, and can even decrease, probably due to slight overfitting.

Results from Experiment 2 show little difference between the larger patch sizes, however there is a drop of 2% in the specificity of inner wall classification for the smallest patch of 20×10. This suggests that a relatively large patch is needed to incorporate useful boundary features.

Results from Experiment 3 show that increasing the number of hidden layers has a marginal detrimental effect on both inner and outer wall specificity. It is possible that there is slight overfitting occurring here, as increasing the layers results in ever more abstract features, with complex features closely resembling those seen in the training samples. Given that the boundary area is diffused, highly abstract and complex features may be too specific for good generalisation. This suggests that a simple NN architecture of one hidden layer is sufficient for boundary detection as simpler, more generalised features are learned.

Based on the results of the three experiments, a NN architecture was chosen for boundary detection in the proposed segmentation framework. For simplicity, the same architecture is used for both inner and outer vessel wall classification. Experiment 1 indicates that 40 nodes in the first hidden layer is a sufficient number in order to produce enough features for generalisation, without over-complicating the problem. Experiment 2 shows that a larger local patch of either 40×20 or 80×40 should be extracted due to their higher specificity for the vessel's inner wall. For simplicity and to reduce computation time, the smaller of the two is chosen. Finally Experiment 3 suggests that a simple architecture of just one hidden layer produces the best specificity results. As a result, the NN boundary detector used for segmentation has a single hidden layer of 40 nodes with local patch extraction of size 40×20.

Figures 5.8 and 5.9 show example boundary detection results of the chosen architecture, for both the inner and outer vessel walls. Both show that the true-positive count is very high with very little gaps in the contour. This is true even for regions with very weak edges. The false-positive classification count is significantly higher than that of the false-negatives, hence the lower specificity values. However, it is clearly shown in these Figures that the vast majority of false-positives are in the immediate vicinity of the ground truth boundary. This is further emphasised by Figure 5.7, which show histogram plots of the false-positive counts relative to their distance to the ground truth boundary.

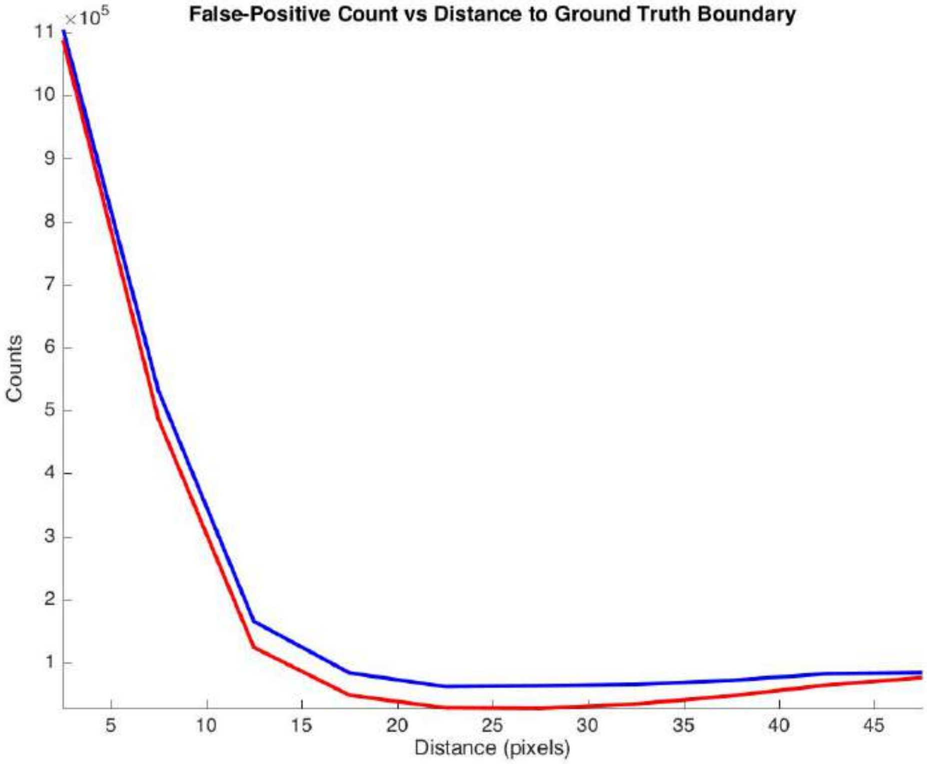


Figure 5.7: Histograms showing the incorrectly classified non-boundary pixels against their distances to the boundary ground truth. Blue: Inner Wall; Red: Outer Wall.

Given that the appearance of the lymphatic vessel's outer wall is diffused, classification errors would be expected in this area. There is no obvious difference in appearance between ground truth boundary pixels and their immediately adjacent neighbours, therefore misclassification here is understandable. In the deformable modelling context, in such cases where the exact boundary location is inevitably difficult to detect, it is conventionally useful to apply additional non-data driven forces such as shape constraints to curb the contour's deformation. Given the location of the misclassified boundary pixels, these results suggest that NNs may be highly suitable for boundary detection in the proposed deformable modelling framework.

Once the boundary detector has been embedded into the segmentation framework, another additional process follows boundary detection and before mesh regularisation. Given the assumption that both the inner and outer vessel walls are tubular in shape, any distant outliers from an ellipse-like shape on any slice are discarded, and are replaced by interpolated contour points.

Given a 2D contour C containing n points, an ellipse is fitted which results in a new contour of n points, C'. Points in both contours are correspondent to each other, which means that point j in C is correspondent to point j in C'. The same method is used to fit the data to an ellipse as was used during initial segmentation (Section 5.3.1). Then, for every point in C, its nearest neighbour in C' is found, and any point which has a nearest neighbour Euclidean distance of more than threshold t is identified as an outlier. If point j in C is considered an outlier, it is then simply replaced by point j in C', leaving a contour close to the object boundary with no distant outliers. In this framework, the outlier distance threshold is $t = 25$ pixels.

This small step becomes important especially when detecting the boundary of the inner wall. In all volumes a valve-like structure can be seen at the centre of the lymphatic vessel, which on some slices can have stronger boundary features than the inner wall itself. Therefore after boundary detection it is likely that many vertices are deformed towards the valve rather than the vessel's inner wall. Using this simple interpolation process however, can easily identify these vertices, and are replaced with new interpolated points.

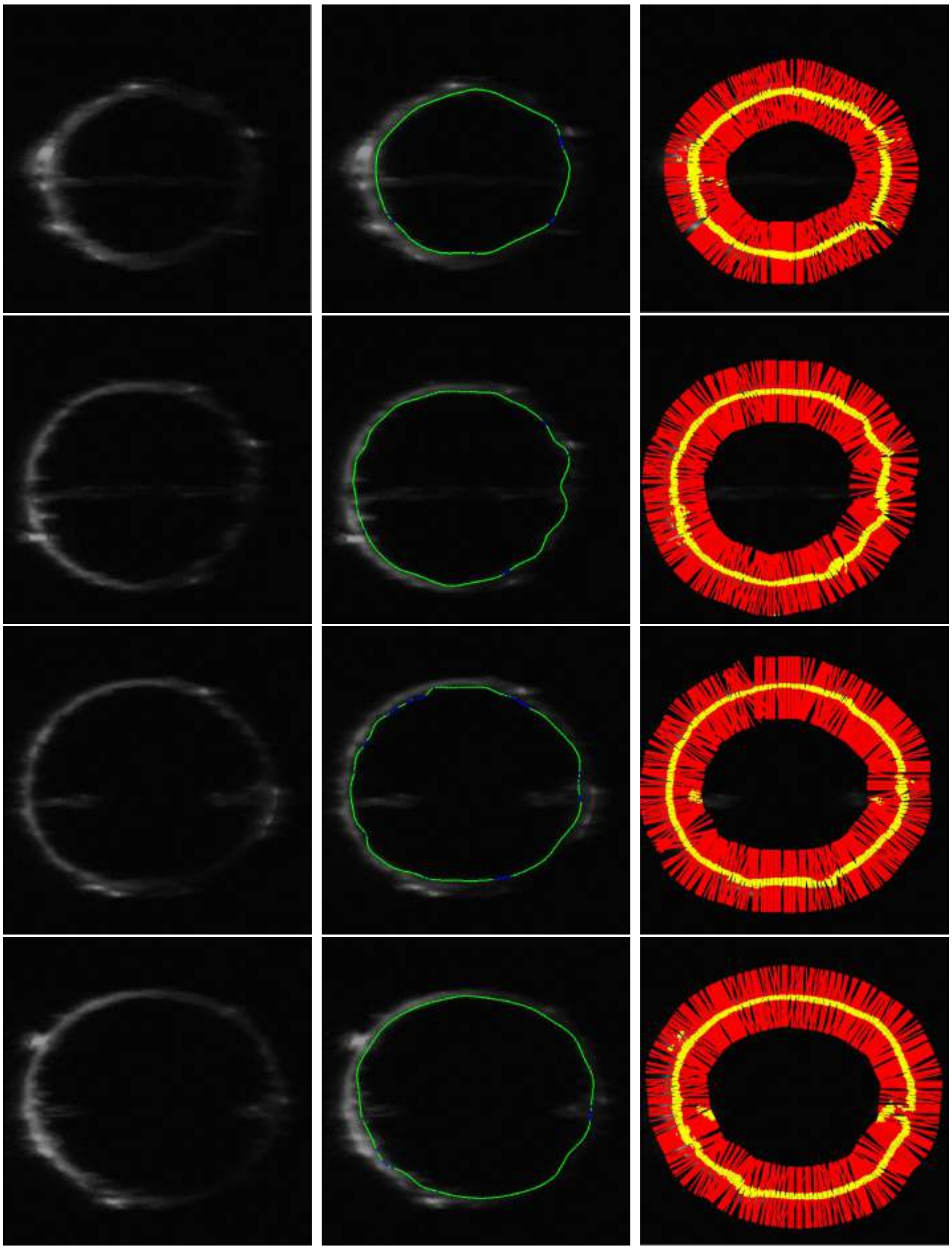


Figure 5.8: Inner wall boundary detection results for one volume, where each row shows the results of a different slice. Left column: 2D image slice. Centre column: Green - correctly classified boundary pixels (true positives); Blue - incorrectly classified boundary pixels (false negatives). Right column: Red - correctly classified non-boundary pixels (true negatives); Yellow - incorrectly classified non-boundary pixels (false positives).

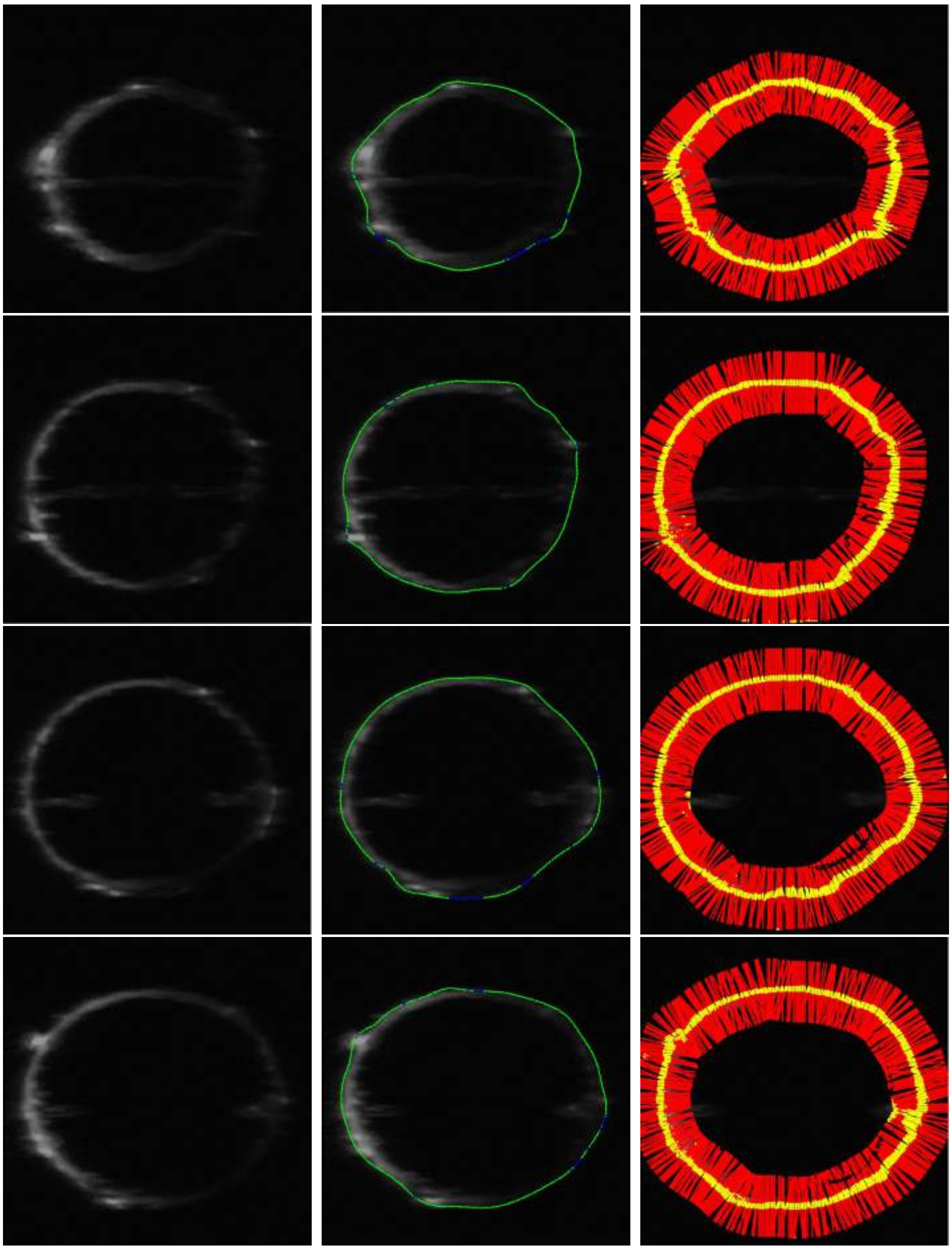


Figure 5.9: Outer wall boundary detection results for one volume, where each row shows the results of a different slice. Left column: 2D image slice. Centre column: Green - correctly classified boundary pixels (true positives); Blue - incorrectly classified boundary pixels (false negatives). Right column: Red - correctly classified non-boundary pixels (true negatives); Yellow - incorrectly classified non-boundary pixels (false positives).

5.3.3 Mesh Regularisation

Before boundary detection, there is an original set of mesh vertices V, and after boundary detection there is now a new set of mesh vertices at different locations V'. As there is no shape restrictions in these components (apart from the length of the boundary detector search path itself), an additional process is needed to preserve the mesh's smooth surface. Here the same B-spline based mesh regularisation is used as in Chapter 4, where a local transformation $T(x, y, z)$ between V and V' is estimated with 3D B-splines. The transformation is then performed on V using free-form-deformation, so that it fits as close as possible to V'. As a result, the smoothness of the transformed mesh is purely a result of the degrees of freedom of the B-splines themselves.

The transformation is estimated in a multi-resolutional procedure by warping an underlying voxel lattice manipulated by control points $\phi^h_{i,j,k}$ of size $[n_x \times n_y \times n_z]$. Readers are referred to Chapter 4, and Equations 4.4, 4.5, 4.6, 4.7 and 4.8 on how to estimate $T(x, y, z)$, and how to optimise the control point locations $\phi^h_{i,j,k}$.

The spacing between $\phi^h_{i,j,k}$, δ, is what determines the degrees of freedom of the transformation. A high δ yields less control points that are sparsely separated. This means that the B-spline interpolation distance between control points is longer, giving the transformation less degrees of freedom. A high δ therefore results in a very smooth mesh surface, however it has less freedom to deform towards areas of high curvature, and therefore the true object boundary. A low δ on the other hand increases the number of control points, making interpolation distances shorter, allowing more degrees of freedom. This results in a less smooth mesh surface, but closer to the true object boundary. A trade-off must therefore be found to allow V to deform as close to the boundary positions as possible (V'), while resulting in a sufficiently smooth surface.

During the iterative deformable modelling, two parameters are changed on every iteration in order to achieve such a trade-off. Firstly, the boundary detector's search path decreases on every iteration, which allows the system to reach convergence more quickly. Secondly, as the amount of possible deformation is reduced at each iteration, it is less likely that the mesh surface will be tangled even if the mesh regularisation has more degrees of freedom. Therefore on every iteration, the value of δ is also reduced. Figure 5.10 shows the effect of the mesh regularisation by comparing it to what the vertices look like after boundary detection.

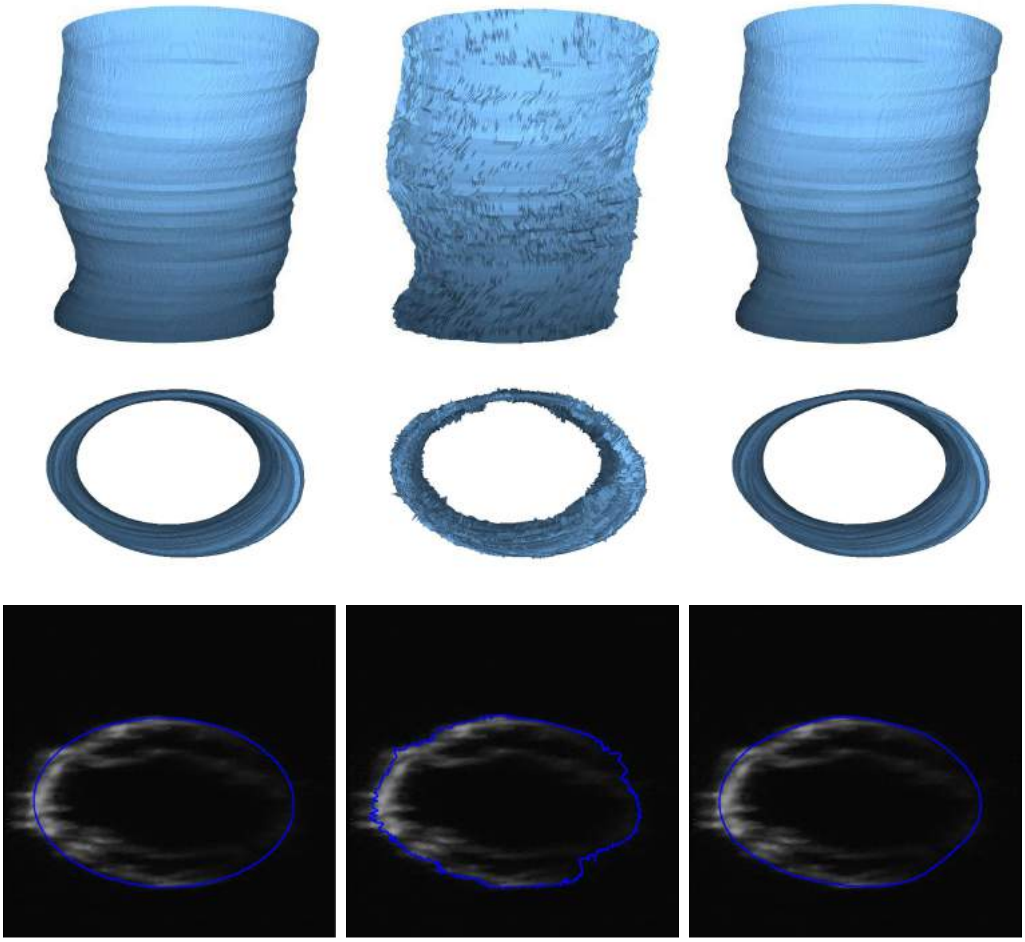


Figure 5.10: Left column: Mesh and corresponding segmentation on a 2D slice of the initial segmentation. Centre column: Mesh and a segmented slice after 2D boundary detection. Right column: Mesh and a segmented slice after 3D mesh regularisation.

5.4 Comparative Analysis and Results

The proposed deformable model-based approach is compared to two additional image-driven approaches, which were implemented and tested on the same dataset. Firstly a simple intensity-based approach was implemented using Haar-like filtering. This was essentially the same filtering method used in Section 5.3.1, but without contour smoothing in the cartesian coordinate system. This allowed comparison with a purely image-driven approach which had no shape regularisation at all. For the remainder of this chapter, this approach is referred to as *intensity-based segmentation.*

Secondly, comparisons were made to another intensity-based approach, but this time with some soft shape constraints. This consisted of the Haar-like filtering in polar coordinates followed by contour smoothing in the cartesian coordinate system. As such this approach is essentially the initial segmentation of the proposed framework, and is refereed to as *initial segmentation* for the remainder of this chapter. Comparing the initial segmentation results with that of the proposed framework shows the quantitative effect of implementing the additional deformable modelling stage. Quantitative and qualitative results were obtained for both of these approaches.

For fair comparison, the results on testing data of the proposed method were only compared to other methods working on the same dataset. As such, it was also possible to make quantitative comparisons to other methods which Essa et al. [75] implemented on the same confocal microscopy images. In this work four segmentation methods were implemented which included an optimal surface approach [155], a simplistic tracking-based approach using the Viterbi algorithm [80], and two additional tracking-based approaches proposed by the authors themselves. Their results clearly showed that the tracking-based approach referred to as *minimum s-excess graph segmentation* produced the best segmentation, and so this result is compared to that of the intensity-based, initial, and proposed segmentations. In doing so it provides a comparison to a segmentation approach that is completely different to the proposed framework. For fair evaluation the error metrics used here [75] were computed in exactly the same way on the proposed method, initial segmentation, and intensity-based results.

Proposed by Essa et al. [75], *minimum s-excess graph segmentation* involves formulating a graph to segment both inner and outer walls simultaneously in polar coordinates. Hand crafted edge features are used for graph cut in each polar image column, while a hidden Markov model was used to track the vessel walls between the columns.

All segmentation experiments were implemented on six $512 \times 512 \times 512$ confocal microscopy images of the lymphatic vessel, and performed with leave-one-out cross-validation. For each volume the inner and outer wall ground truth meshes were obtained by labelling every tenth slice, and manually defining mesh edge connections (as is described in Section 5.3.1). Inner and outer wall segmentations were obtained independently, and so the results were also evaluated independently.

All evaluation metrics were computed consistently with Essa et al. [75], for fair comparison. Evaluation was performed on a 2D slice-by-slice basis in polar coordinates. The point-to-mesh distance (PMD), Hausdorff distance (HD), area overlap (AO), specificity and sensitivity were all computed as described in Appendix A.

The proposed framework achieved convergence in under four iterations, in all runs on testing data. The boundary detector search path length was initialised at 20 pixels and was halved on every iteration, while the FFD control point spacing also decreased. This combination yielded fast convergence. The framework achieved point-to-mesh distances of 1.6 ± 0.1 and 1.5 ± 0.1 pixels for the inner and outer vessel walls, respectively, while the Hausdorff distance measures were 5.8 ± 0.5 and 5.4 ± 0.4. This shows that the segmentation results showed very close resemblance to the ground truth, with no distant outlying regions in both inner and outer wall segmentations. The high scores of the area overlap, sensitivity and specificity for both inner and outer walls also show high accuracy, and can be seen in Tables 5.4 and 5.5. In addition, the standard deviations of all evaluation metrics for the proposed framework show that the variation in the results are low, indicating a system of high precision.

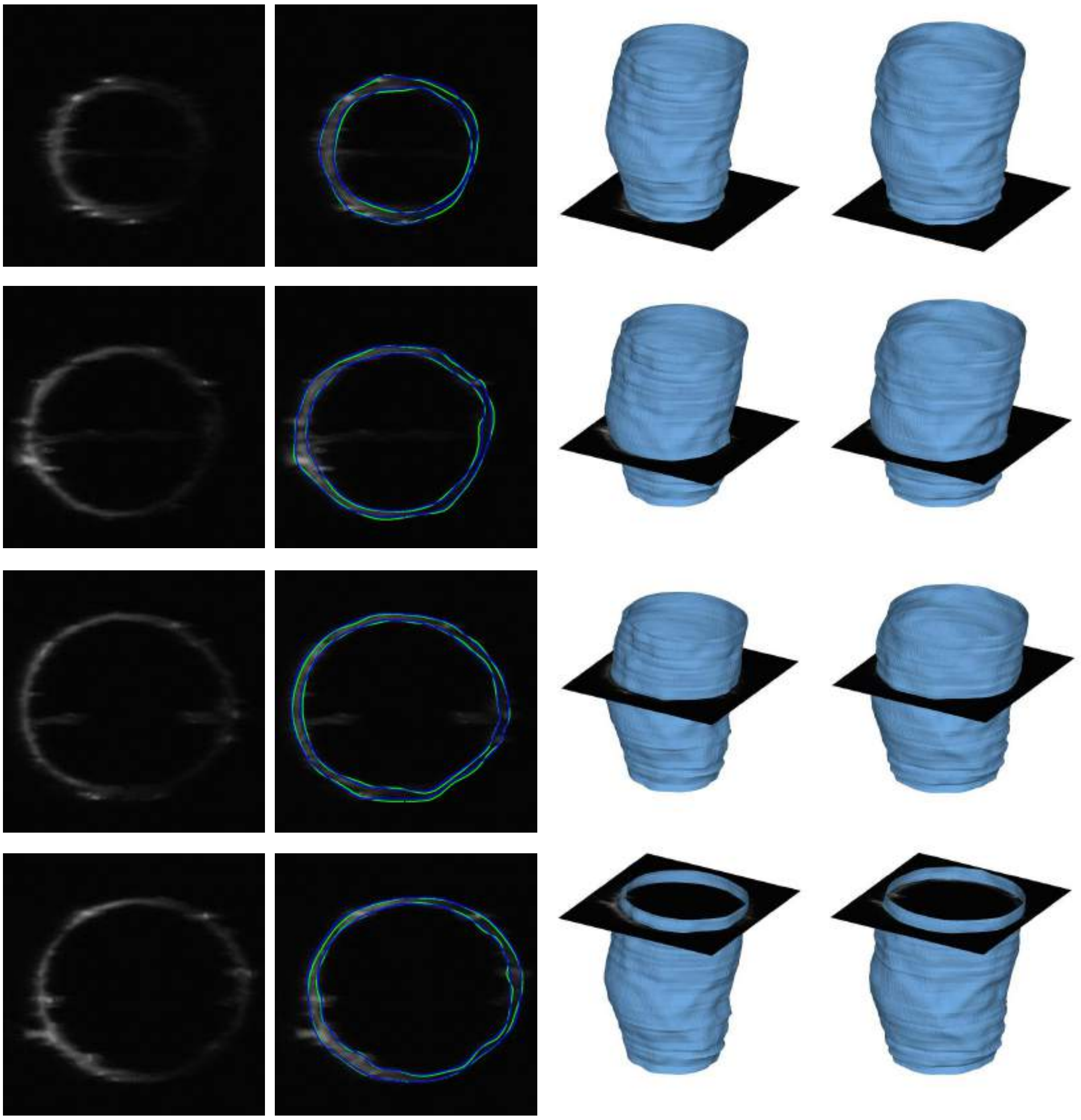


Figure 5.11: Segmentation results for test volume 1. From left to right; 1st column: Image slices. 2nd column: Inner and outer wall segmentation results. Green contours are the ground truth and blue contours are the result. 3rd column: Resulting inner wall mesh with corresponding slices. 4th column: Resulting outer wall mesh with corresponding slices.

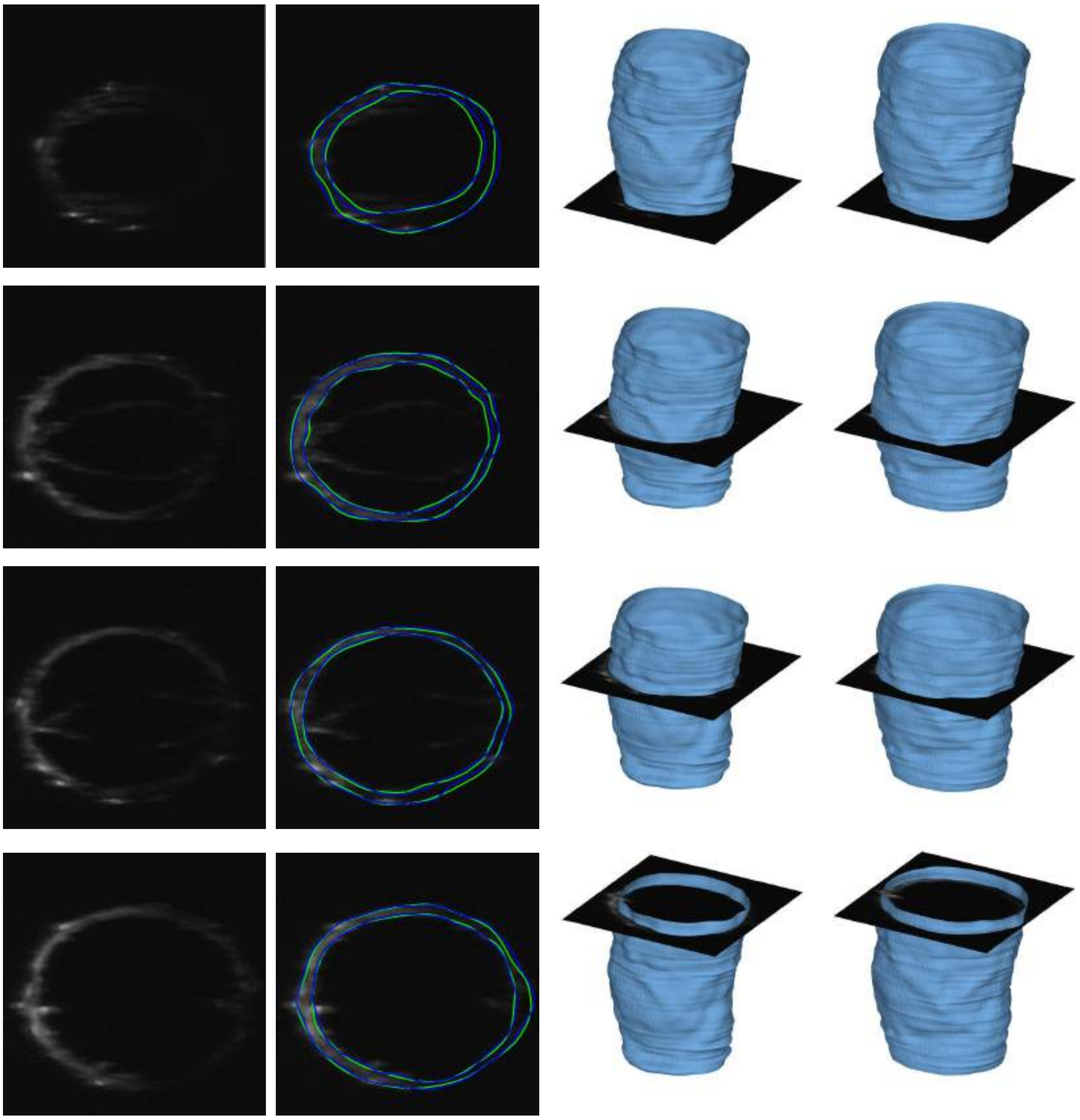


Figure 5.12: Segmentation results for test volume 2. From left to right; 1st column: Image slices. 2nd column: Inner and outer wall segmentation results. Green contours are the ground truth and blue contours are the result. 3rd column: Resulting inner wall mesh with corresponding slices. 4th column: Resulting outer wall mesh with corresponding slices.

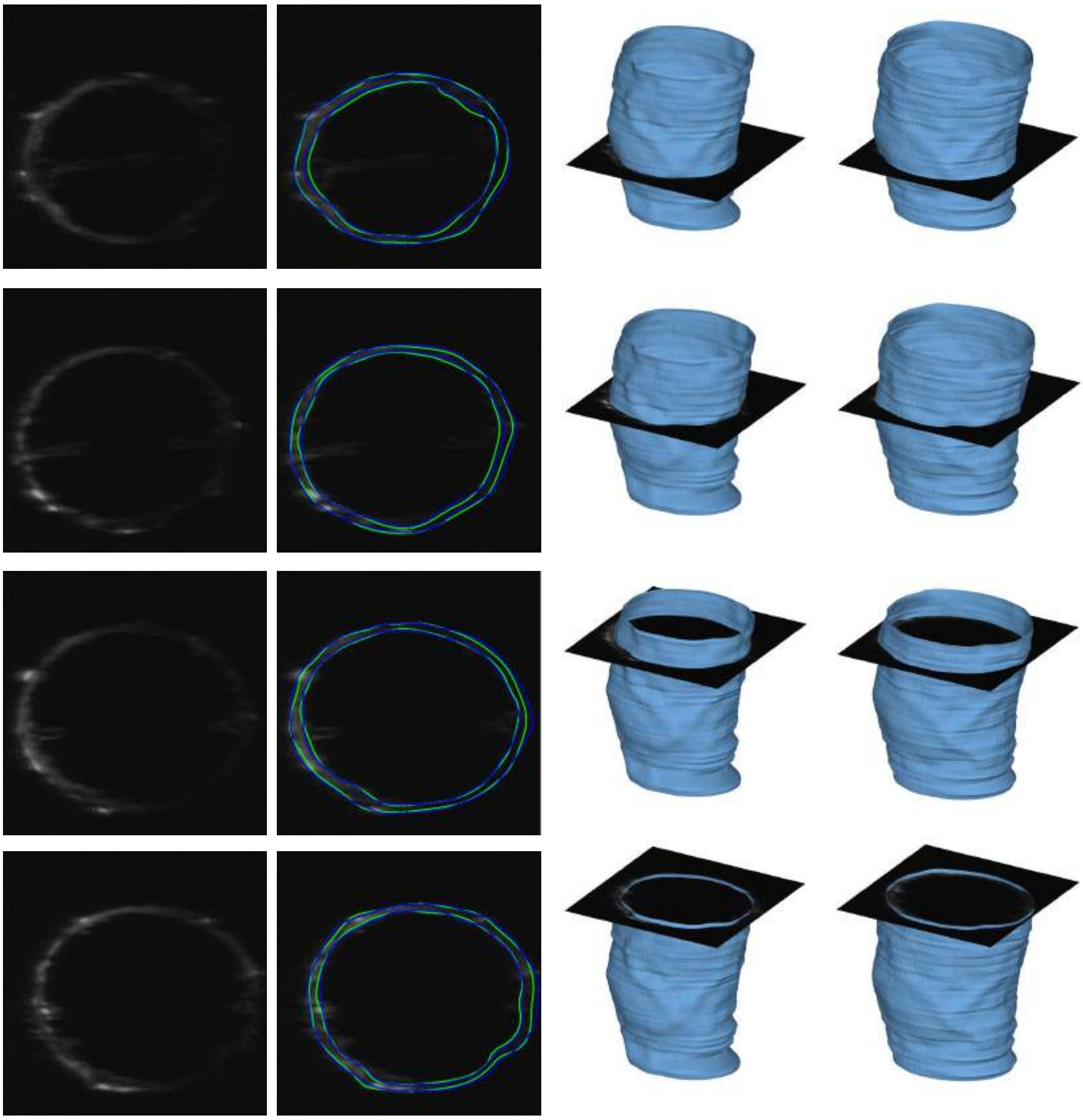


Figure 5.13: Segmentation results for test volume 3. From left to right; 1st column: Image slices. 2nd column: Inner and outer wall segmentation results. Green contours are the ground truth and blue contours are the result. 3rd column: Resulting inner wall mesh with corresponding slices. 4th column: Resulting outer wall mesh with corresponding slices.

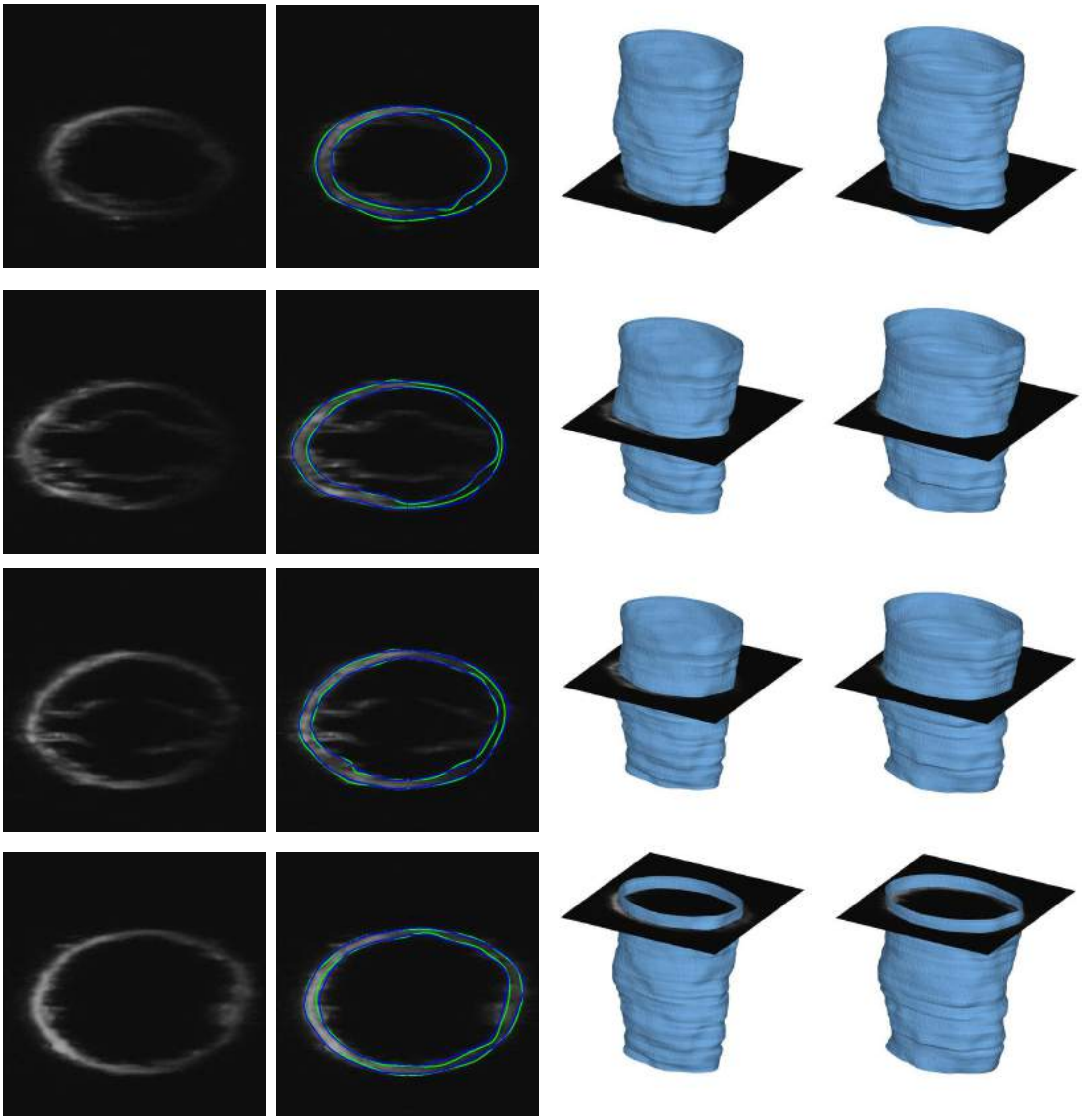


Figure 5.14: Segmentation results for test volume 4. From left to right; 1st column: Image slices. 2nd column: Inner and outer wall segmentation results. Green contours are the ground truth and blue contours are the result. 3rd column: Resulting inner wall mesh with corresponding slices. 4th column: Resulting outer wall mesh with corresponding slices.

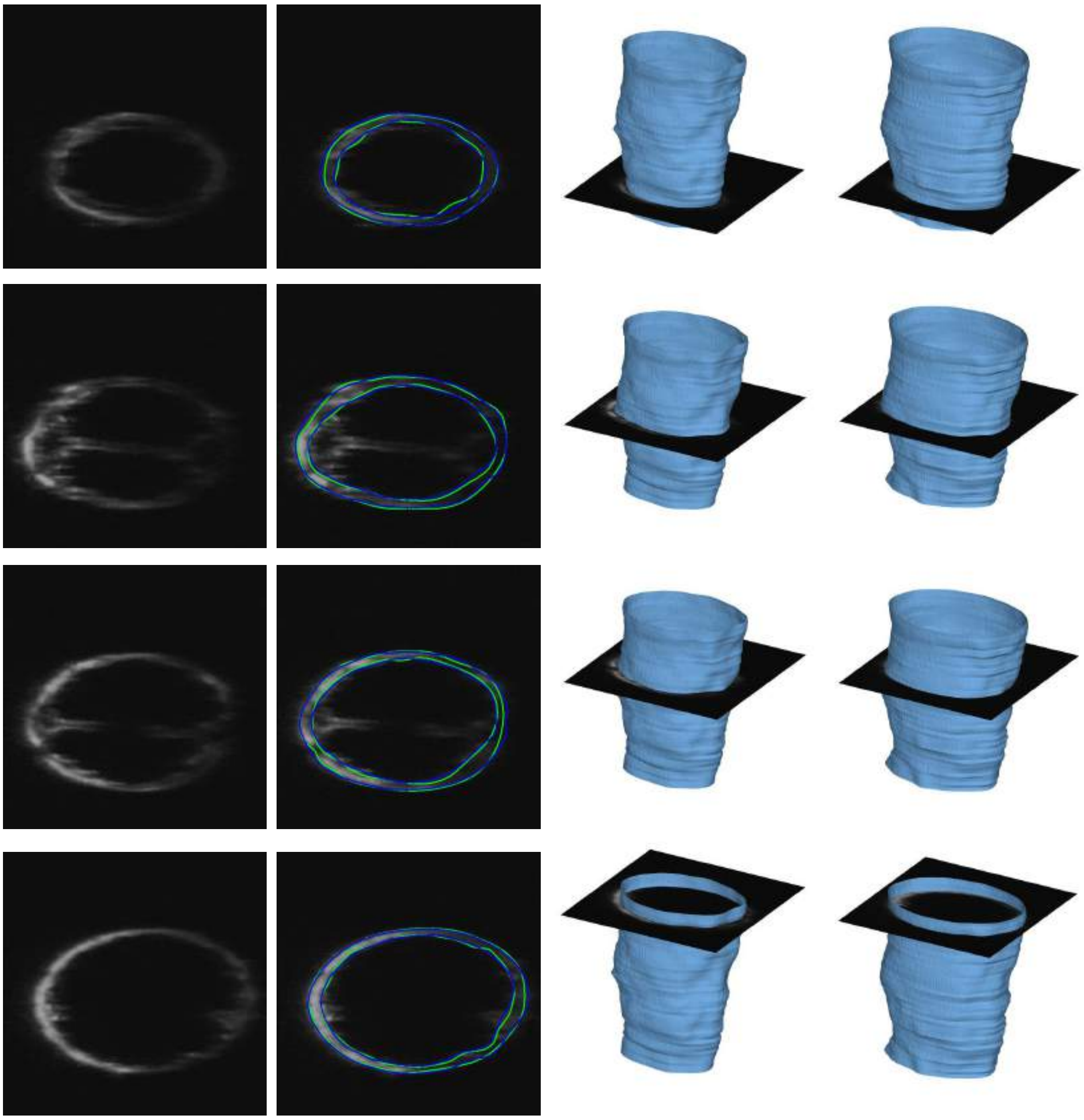


Figure 5.15: Segmentation results for test volume 5. From left to right; 1st column: Image slices. 2nd column: Inner and outer wall segmentation results. Green contours are the ground truth and blue contours are the result. 3rd column: Resulting inner wall mesh with corresponding slices. 4th column: Resulting outer wall mesh with corresponding slices.

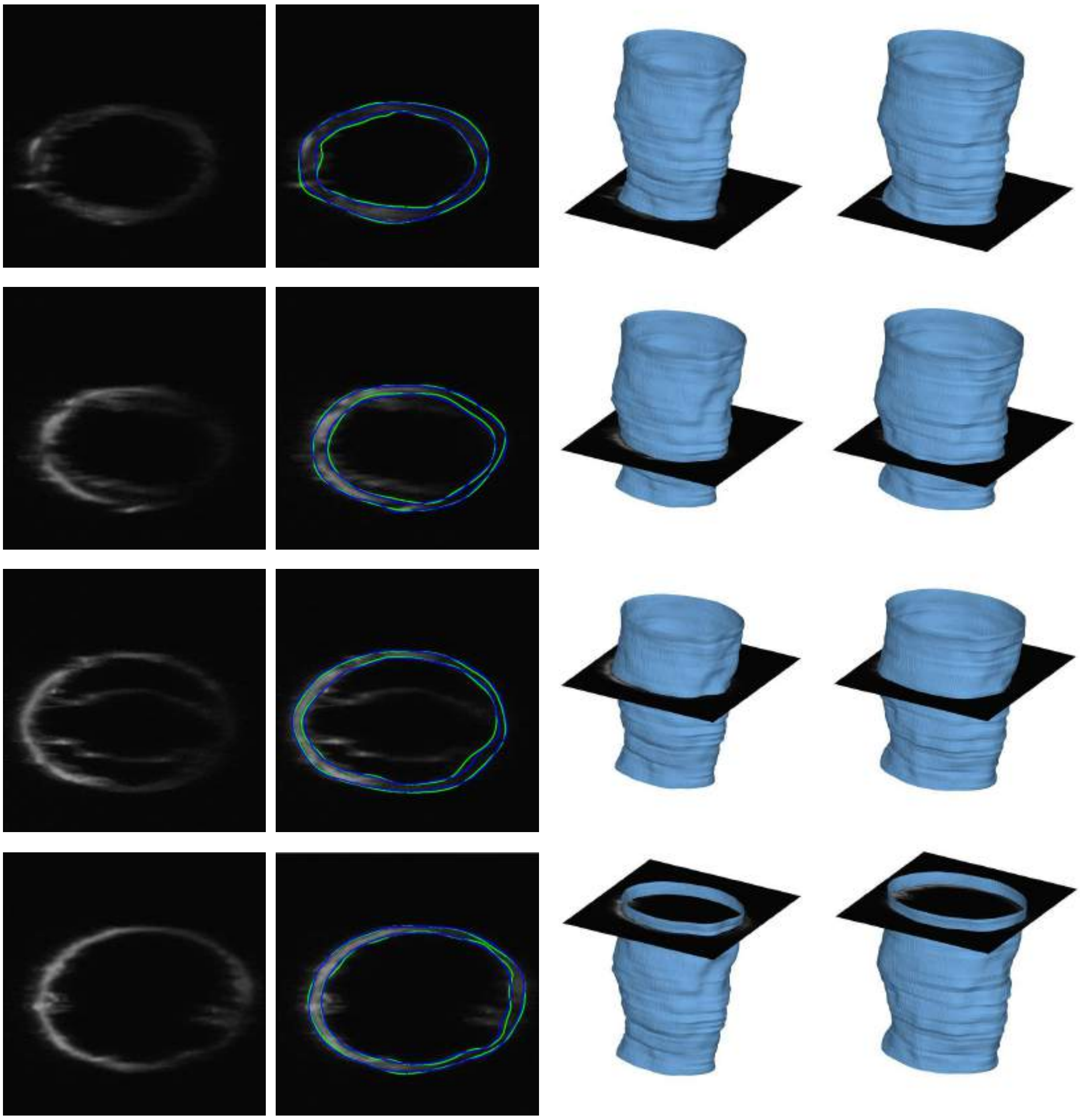


Figure 5.16: Segmentation results for test volume 6. From left to right; 1st column: Image slices. 2nd column: Inner and outer wall segmentation results. Green contours are the ground truth and blue contours are the result. 3rd column: Resulting inner wall mesh with corresponding slices. 4th column: Resulting outer wall mesh with corresponding slices.

Method	**PMD (voxels)**	**HD (voxels)**	**AO (%)**	**Sens. (%)**	**Spec. (%)**
Min S-Excess Graph [75]	3.1 ± 1.9	9.8 ± 4.3	95.5 ± 3.2	96.9 ± 3.1	99.2 ± 0.8
Intensity-based	5.9 ± 0.8	48.5 ± 7.1	92.4 ± 1.5	93.0 ± 1.4	99.6 ± 0.1
Initial Segmentation	2.9 ± 0.4	9.3 ± 0.9	96.4 ± 0.4	97.0 ± 0.4	99.6 ± 0.1
Proposed Framework	$\mathbf{1.6 \pm 0.1}$	$\mathbf{5.8 \pm 0.5}$	$\mathbf{98.0 \pm 0.4}$	$\mathbf{99.1 \pm 0.4}$	99.2 ± 0.2

Table 5.4: Inner wall quantitative results comparison on test data.

Method	**PMD (voxels)**	**HD (voxels)**	**AO (%)**	**Sens. (%)**	**Spec. (%)**
Min S-Excess Graph [75]	2.0 ± 0.8	7.4 ± 3.1	97.6 ± 1.0	98.7 ± 1.1	99.1 ± 0.6
Intensity-based	4.5 ± 1.4	46.9 ± 8.1	95.0 ± 1.8	96.9 ± 1.4	98.3 ± 0.8
Initial Segmentation	1.7 ± 0.2	5.7 ± 0.3	98.2 ± 0.1	99.2 ± 0.1	99.0 ± 0.3
Proposed Framework	$\mathbf{1.5 \pm 0.1}$	$\mathbf{5.4 \pm 0.4}$	$\mathbf{98.4 \pm 0.1}$	$\mathbf{99.2 \pm 0.02}$	$\mathbf{99.2 \pm 0.03}$

Table 5.5: Outer wall quantitative results comparison on test data.

Figures 5.11, 5.12, 5.13, 5.14, 5.15 and 5.16 show qualitative results of all six test volumes. These figures show how unclear parts of the vessel walls are in the data, and also show the close correlation between the resulting contours and the ground truth. Smooth resulting mesh surfaces can also be seen, with no tangled or extremely faceted mesh faces.

Of the image-driven approaches implemented in this experiment, the proposed method produced the best segmentation results. Tables 5.4 and 5.5 show that the proposed method produced the lowest PMD, HD, AO, specificity and sensitivity results compared to the intensity-based and initial segmentations. Figures 5.17 and 5.18 compare the qualitative slice segmentations and mesh results for the inner and outer walls of the image-driven approaches.

It is immediately noticeable that the simplistic intensity-based segmentation produced significantly higher PMD and HD errors compared to both the initial segmentation and the proposed framework. This suggests that large regions of both walls deviated significantly from the ground truth, which is clearly shown in Figure 5.17. The outer wall seems to deviate away from the true boundary on the right hand side of the image slices where the vessel becomes dimmer. Meanwhile the boundary detector for the inner wall has caused vertices to deform towards the valve at the centre of the vessel, instead of the wall itself. The lack of any shape regularisation means that the process is purely image-driven, and is unsuitable for such data where the vessel walls are not always prominent. This lack of regularisation is also clearly noticeable in its mesh appearance in Figure 5.18, where the mesh surface is highly faceted.

The initial segmentation results are significantly better, with the PMD being almost half that of the simplistic intensity-based method, and the HD is dramatically reduced. By fitting each slice contour to an ellipse the majority of *correctly* deviated vertices have forced the *incorrectly* placed vertices nearer the boundary walls. This simple shape regularisation approach has also had dramatic effects on the smoothness of the mesh surface. However, as each contour has been fitted to a smooth ellipse, the approach does not allow enough degrees of freedom to precisely reach the areas of high curvature on the boundary.

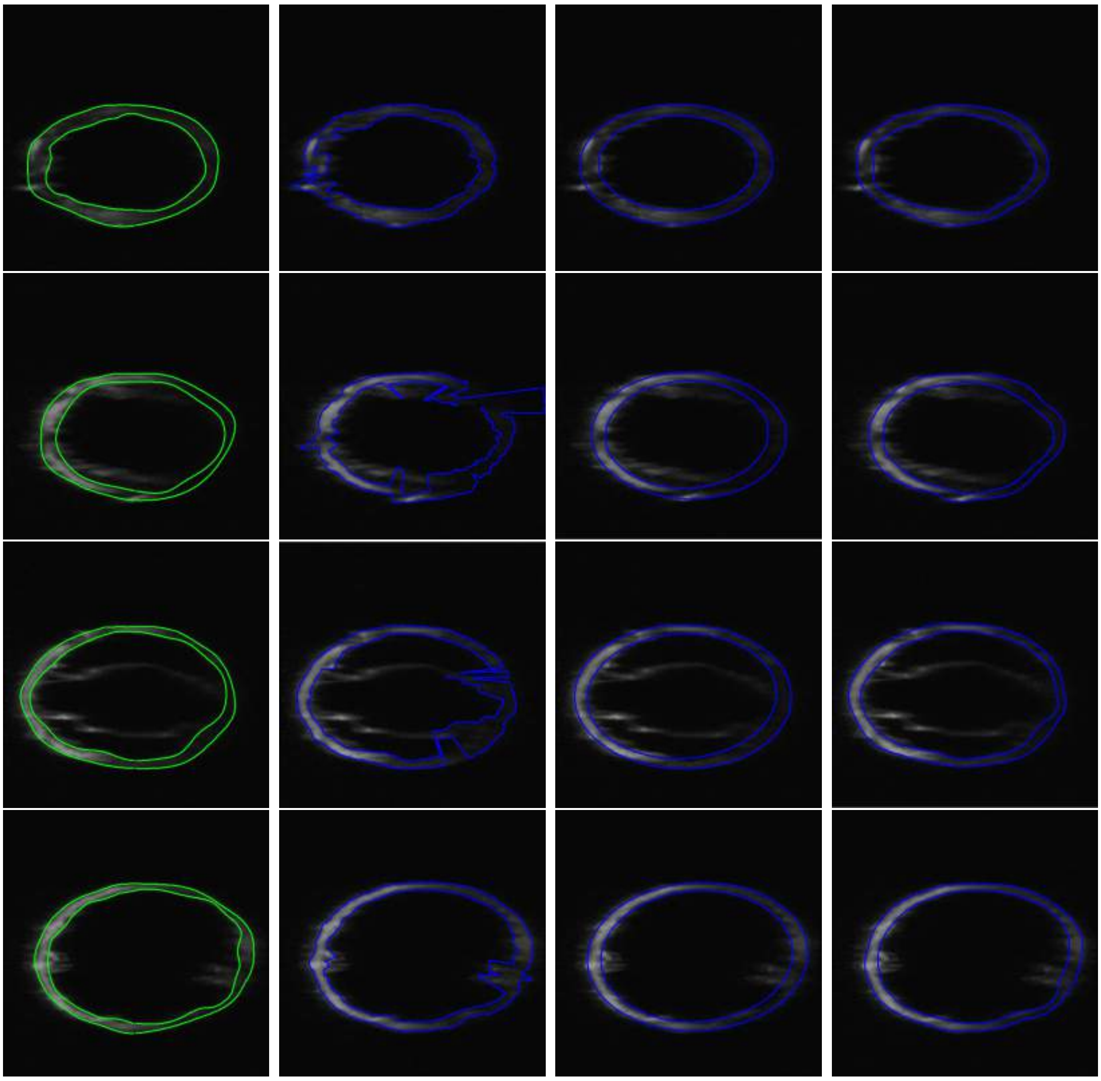


Figure 5.17: Comparison results on image slices. From left to right; 1st column: Ground truth. 2nd column: Intensity-based segmentation. 3rd column: Initial segmentation. 4th column: Proposed framework.

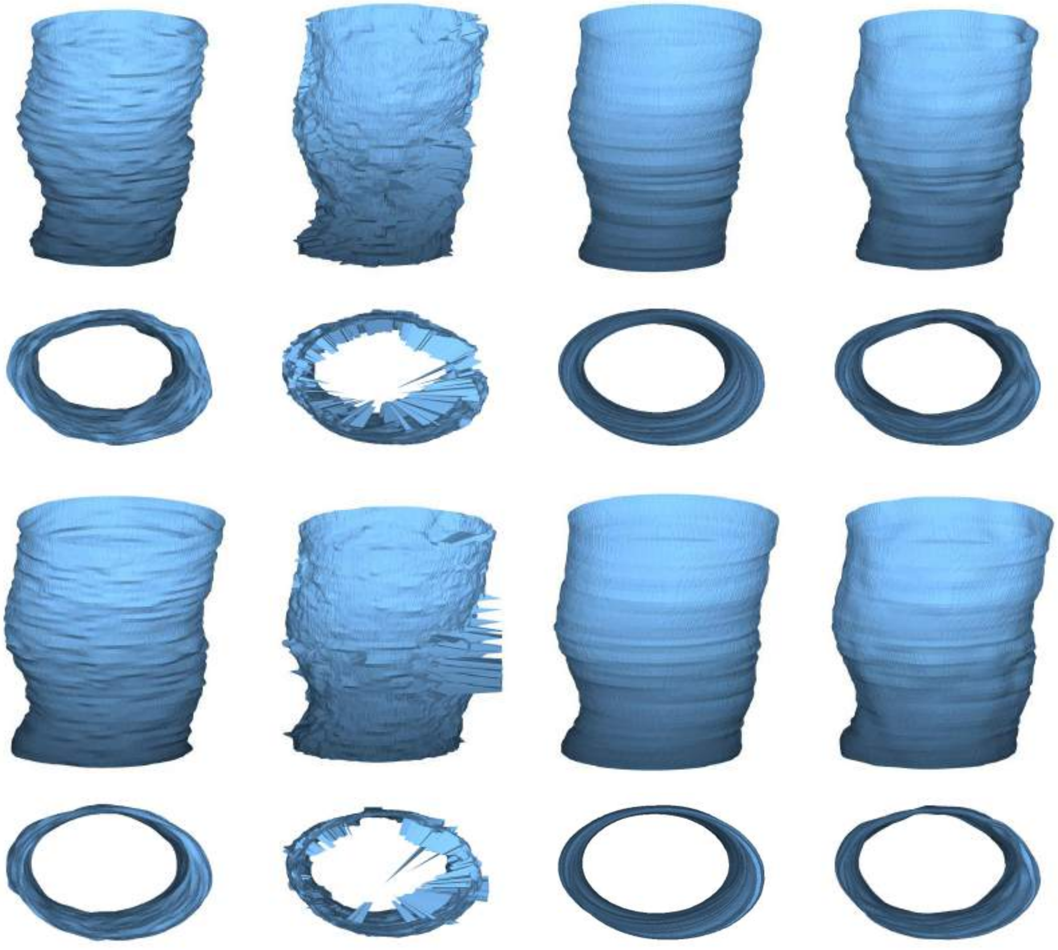


Figure 5.18: Comparison mesh results. The top two rows show the inner vessel wall mesh. The bottom two rows show the outer wall mesh. From left to right; 1st column: Ground truth. 2nd column: Intensity-based segmentation. 3rd column: Initial segmentation. 4th column: Proposed framework.

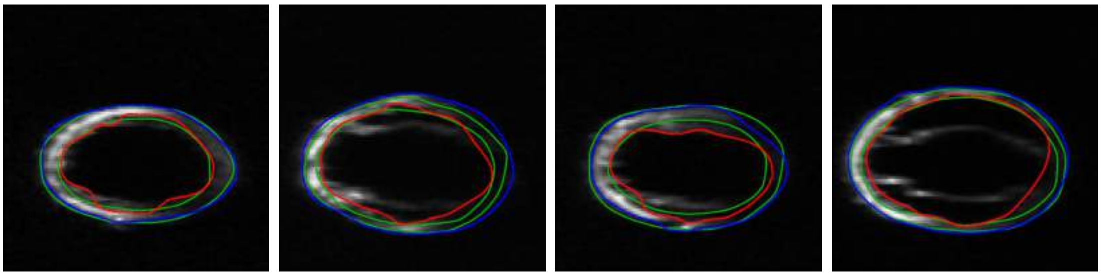


Figure 5.19: Example results from the minimum s-excess graph method provided by Essa et al. [75]. Green: ground truth; Red: inner wall segmentation; Blue: outer wall segmentation.

The results from the proposed method shows that an additional deformable modelling process is necessary after initial segmentation. The evaluation results are lower still than the initial segmentation, especially for the inner wall where segmentation errors have significantly fallen. The iterative boundary detector allows deformation towards areas of high curvature, which is represented by both the low PMD and HD errors. Meanwhile the iterative mesh regularisation maintains the mesh's smooth surface, which can be seen in Figure 5.18.

Compared to the minimum s-excess graph method, the proposed method still produced better segmentation results. The tracking-based method produced PMD and HD metrics that were higher for the outer wall, and almost double that of the proposed method for the inner wall. This is also reflected in the AO, specificity and sensitivity metrics.

In addition, Figure 5.19 shows example segmentation slices provided by Essa et al. [75]. From these images it can be seen that the outer wall segmentation falters towards the right hand side of the slice, where the vessel's appearance is more obscure. It is this region that increases the overall PMD and HD errors. The largest error in this method however is in the inner wall, which is highlighted by comparing its evaluation metrics with that of the outer wall. Similar to the intensity-based approach, the inner wall segmentation deviates towards the valve-like structure at the centre of the vessel, increasing the PMD and HD errors. This indicates that the method has some difficulty in distinguishing between the vessel wall boundary and the boundary of the valve, while also struggling to identify edges with varying degrees of contrast. This may be down the tracking model of the method itself, however a likely cause is the hand-crafted edge features used for emission probability. This being the case it would show that using learned features from algorithms such as NN has an advantage for edge detection in such data.

5.5 Summary

In this chapter a fully automatic deformable modelling method has been presented for the segmentation of 3D lymphatic vessels in confocal microscopy images. A similar bottom-up, image-driven framework to that presented in Chapter 4 was used, which included a learning-based boundary detector and mesh regularisation for shape preservation. A simple intensity-based initial segmentation was first adopted which was followed by a deformable modelling system. A neural network was used here for boundary detection, which allowed suitable features to be learned instead of hand-crafting them. This proved particularly useful as choosing features for edges with varying degrees of contrast is not a trivial task.

The proposed initial segmentation produced a starting mesh which was close enough to the boundary in order for accurate deformable modelling to follow. The NN-based boundary detector was then able to accurately detect both vessel walls, which proved it was capable of detecting edges of highly varying contrasts. Mesh regularisation was also necessary in order to obtain smooth vessel surfaces.

Comparisons were also made to an intensity-based segmentation, the initial segmentation, and a tracking-based approach. By comparing the intensity-based and initial segmentation results, it was shown that contour smoothing was necessary in order for the inner wall segmentation to avoid the vessel's valve. In addition by comparing the initial segmentation results to those of the proposed framework, it was clear that the deformable modelling stage was necessary in order to provide the segmentation with enough degrees of freedom to converge closer to the boundary walls.

Chapter 6

Conclusions and Future Work

Contents

6.1 Conclusions

The aim of this work was to develop a fully-automatic deformable modelling method to accurately and efficiently segment the aortic root in 3D CT images, and to extend the method to delineate vascular structures. A framework was presented to deform the mesh towards the object boundary, and consisted of boundary detectors using machine learning algorithms, and a 3D mesh regularisation technique using B-spline-based local transformation. This image-driven framework was also applied to confocal microscopy images to segment the lymphatic vessel, showing that with minor adjustments the framework is flexible enough to work on multiple datasets. Furthermore at the boundary regions of interest, both the CT and confocal microscopy images had highly obscured appearances with inconsistent contrast. This shows that although there are no strong high-level shape constraints placed on the model, it is still able to delineate difficult boundaries.

For aortic root segmentation, the framework was initialised by using machine learning for pose estimation through MSL. Two implementations were presented with both hand-crafted features (RF-MSL) and automatically learned features (NN-MSL). Results for both were comparable, which shows that the deep learning approach learned sufficiently abstract features for pose estimation in large volumes. Furthermore, novel multi-resolution pooling was used with NN-MSL to significantly reduce the network sizes and subsequently their training times. Its use of multi-resolution intensity extraction on an irregular grid allowed information from both large and small-scale structures to be used for classification.

Due to the complex shape of the aortic root, shape constraints for an SSM was applied for shape preservation. However in contrast to traditional ASM-based segmentation, these top-down shape constraints were applied non-iteratively. This was to preserve a valve-like shape during the initial large deformations, but to rely on the data for the following iterations to allow sufficient adjustability. Given the use of an SSM, a technique was also developed to reduced manual work for its preparation by automatically finding point correspondences across a set of aortic root shapes. The mesh-based similarity metric allowed more accurate local transformation between two shapes when compared to an image-based metric. This allowed more accurate correspondences to be identified for SSM generation, and dramatically reduced model training time.

With some changes, the framework was also applied to vessel segmentation. MSL pose estimation was replaced with an initial intensity-based filtering segmentation, which generated a mesh close to the ground truth boundary. 2D Learning-based boundary detectors were then used to drive the model and 3D mesh regularisation was used for soft shape preservation. Non-iterative SSM constraints were not applied here due to the nature of the vessel's tubular structure, ensuring that the method was strongly image-driven. NNs were again used for boundary detection, and showed good accuracy for obscured vessel walls.

The aortic root segmentation results were comparable to the ground truth, showing good quantitative and qualitative results. The framework's implementation with manual preparation, which included manual labelling for SSM and RF-MSL, produced the best quantitative results compared to other deformable model methods. The automated method consisting of automated SSM generation and NN-MSL showed promise, with comparable quantitative results to the state-of-the-art and smooth regularised mesh surfaces. It is thought that improving the position estimator in NN-MSL at full resolution would improve these results further. The lymphatic vessel segmentation results also outperformed the state-of-the-art, which was a method incorporating strong high-level constraints. The proposed low-level approach was able to deal with inconsistent contrast in the vessel walls, and avoided convergence at the valve-like structure.

The presented deformable modelling framework is reliant on accurate pose parameter estimation, however given the good pose estimations from RF-MSL and NN-MSL, this yielded good segmentation results. This reliance is clear from analysing the aortic root MSL implementations in Chapter 4. Despite the NN-MSL implementation producing similar orientation estimation and boundary detection, its poorer position estimation caused it to yield slightly poorer quantitative segmentation results. Furthermore it was shown that using the ground truth pose for initialisation yielded better overall segmentation than both RF and NN implementations. As such it can be deduced that any improvement to the pose estimation can significantly improve segmentation.

For the lymphatic vessel segmentation in Chapter 5, MSL pose estimation was replaced with an initial intensity-based segmentation. This was possible as there were no other anatomical structures in the confocal microscopy images. However, should the framework be used to segment vascular structures in other imaging modalities, such as CT or MRI, then an MSL-like approach would be necessary. It is thought that the performance of this pose estimation would be similarly critical to the segmentation. Tubular-like structures such as the vessels are simpler in shape than the valve-like aortic root, and so it is assumed that it would be easier to detect. However, the challenge now would be to detect the correct tube-like structure, and being able to distinguish between arteries and veins etc. Information from more global structures might be necessary, and the proposed multi-resolution pooling would be suitable for extracting such features.

The framework also works well on the condition that the initial model is not too dissimilar to the object of interest. The B-spline-based mesh regularisation essentially interpolates the initial mesh to look like the deformed mesh after boundary detection. If the initial and deformed meshes are significantly different then the resulting regularised mesh might be somewhere between the two, and not appear like the desired object at all. As such the proposed B-spline mesh regularisation can be thought of as mesh refinement.

Despite some limitations, given that pathological structures often do not have topology changes, and given a sufficient number of training images and meshes, the presented framework can be used for segmenting many anatomical structures. The NN-MSL and NN boundary detection is capable of pose estimation and driving the model without hand-crafting features specific to the object of interest. This not only means that such classifiers may be treated as *black boxes*, but training time is accelerated as the NNs learn the features capable of such classification.

It is also assumed that the B-spline mesh regularisation is capable of regularising most mesh shapes, given that the initial shape is not too dissimilar and has the same topology as the deformed shape.

6.2 Contributions

The contributions of this thesis are summarised as follows:

1. Non-iterative shape priors. High-level shape constraints were applied non-iteratively for an element of shape preservation after large deformation. After initial segmentation the image-driven forces were emphasised for flexible and accurate segmentation. The advantages of this were shown in the aortic root segmentation in Chapter 4. Applying iterative shape priors in a modified ASM implementation became too restrictive for accurate segmentation, and the proposed framework produced significantly lower quantitative results. Furthermore, the proposed framework produced much more regularised surfaces than the state-of-the-art method with no high-level shape priors at all [288]. This showed that some high-level constraint is necessary for more complex shapes such as valves, and applying them non-iteratively produced the best results.

2. Efficient model training. Automatic SSM generation and deep learning components were combined to dramatically reduce manual work and model preparation time. The clear advantage of automatic SSM generation is its avoidance of extensive manual labelling, and it was shown in Chapters 3 and 4 that identifying automated correspondence yielded comparable overall segmentation to that with manual SSM constraints. Similarly, NNs learn abstract features for suitable classification, which avoids the manual work of hand-crafting such features. The learned features are capable of producing comparable segmentation results, and in their current state are capable of estimating similar object orientation and better object boundary detection compared to the hand-crafted features.

3. Mesh-based similarity metric. An SSM was generated by automatically finding dense vertex correspondence using local transformations. Chapter 3 shows that the mesh-based metric improved local transformation estimation compared to an image-based metric in noisy images. Furthermore, Chapter 4 shows that the automatically generated SSM yielded comparable segmentation results to the manually generated SSM, showing that the current approach is a good replacement for manual labelling.

4. NN-MSL with Multi-Resolution Pooling. Deep learning was proposed for pose estimation which was capable of learning sufficiently abstract features. In doing so it dramatically reduced the manual work needed for hand-picking features suitable for identifying the aortic root's pose. Multi-resolution pooling was also integrated here, which significantly reduced the size of network architectures, while still extracting information of global and local anatomical structures.

6.3 Future Work

6.3.1 NN-MSL Extension

There are several aspects of the segmentation frameworks that can be further developed. Firstly, the NN-MSL pose estimation system can be built upon. For completeness, a further NN-based estimator with multi-resolution pooling can be used for estimating the object's local scale. Although not needed for the case of the aortic root as the local scale variance was low, for other anatomical objects this often is not the case, and some estimation is necessary. Large local patches of different scales can be extracted, and multi-resolution pooling can be applied in the same way as described in Chapter 4. The NN may then classify which scale has the highest detection score, and the scale with highest score is taken as the local scale estimation. Figure 6.3.1 illustrates this, and readers are referred to Chapter 4 for a theoretical description.

Secondly, at its current stage, the maximum image resolution where the ob-

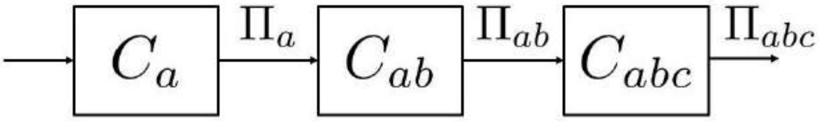


Figure 6.1: An illustration of 3D MSL with scale estimation. C_a is the position estimator which finds high scoring (x, y, z) hypotheses, Π_a. C_{ab} finds the highest scoring $(x, y, z, \omega, \phi, \theta)$ hypotheses Π_{ab}, and finally C_{abc} finds the highest scoring $(x, y, z, \omega, \phi, \theta, S_x, S_y, S_z)$ hypothesis Π_{abc}.

ject's position can be estimated with NN-MSL is equal to the multi-resolution pooling layer with the highest resolution. The estimation is then up-scaled to full resolution. Therefore, it is proposed that an additional NN be used after the current estimator to estimate the position at full resolution. It is believed that NN inputs can be extracted from a local patch in full resolution in such a case.

Such an MSL system may be implemented as follows. 1) NN with multi-resolution pooling used to estimate the object's position within the entire image domain. The estimation resolution is the highest resolution allowed by the multi-resolution pooling, where large patches are extracted to incorporate global structure information. Implemented as described in Chapter 4. 2) An additional NN to estimate position at full resolution, trained with inputs from smaller local patches. High resolution local structure information used to fine-tune the estimation of the previous estimator to full resolution accuracy. Both positive and negative training samples would be taken within close proximity of the ground truth. 3) NN with multi-resolution pooling for orientation estimation, as described in Chapter 4. It is envisaged that improvement to the MSL position estimator would yield significant overall segmentation improvement.

6.3.2 CNN-MSL

Furthermore, it is believed that CNNs would be suitable for pose estimation, and that CNN-MSL be proposed. Conventional CNNs convolves a spatial kernel over an image, and uses the back-propagation algorithm to adjust the appearance of the kernels. These kernels are essentially features which are *learned* to provide suitable classification. Given that CNNs are highly suited to image recognition due their spatial invariance, they would be suitable for recognising local patches at different locations and orientations in the large volumes. However, it is currently expected that information from both global and local structures are needed for accurate estimation, which given a 3D CNN architecture, would require a very large number of parameters to be optimised. Multi-resolution pooling was used to alleviate this problem for NNs (Chapter 4), which extracted intensities from a local patch to form an alternative patch representation on an irregular grid. However, CNNs assume that the topology of the input images are consistent so that conventional convolutional operations can be employed. Applying a conventional CNN architecture with multi-resolution pooling cannot be implemented, as spatial convolutional operators cannot be performed on irregular grid structures.

A possible solution would be to essentially train two (or more) CNNs separately, each with different kernel sizes and resolutions. This would ensure that both global and local structures are represented in the model. These CNNs would then have to merge at the fully connected layer, making the system singular, rather than sequential. Figure **blah** illustrates this idea. Implementing such a system would require optimising many parameters, and extensive work would be required to implement a variant of the backpropagation algorithm, as propagating the errors backwards through CNNs that are essentially "split" from each other is not straightforward. However, if implemented correctly, it is hoped that all pf the benefits that CNNs bring to image recognition can be utilised for MSL pose detection.

6.3.3 CNN-BD

It is also believed that 2D CNNs are suitable for the boundary detection (CNN-BD) of the lymphatic vessel wall. In the current framework, intensities from local patches of 40×20 are extracted and inputted as individual nodes in a NN. A local patch of this size is ideal for CNN image recognition, where convolutional layers can generalise the patch, and pooling layers can reduce the number of parameters.

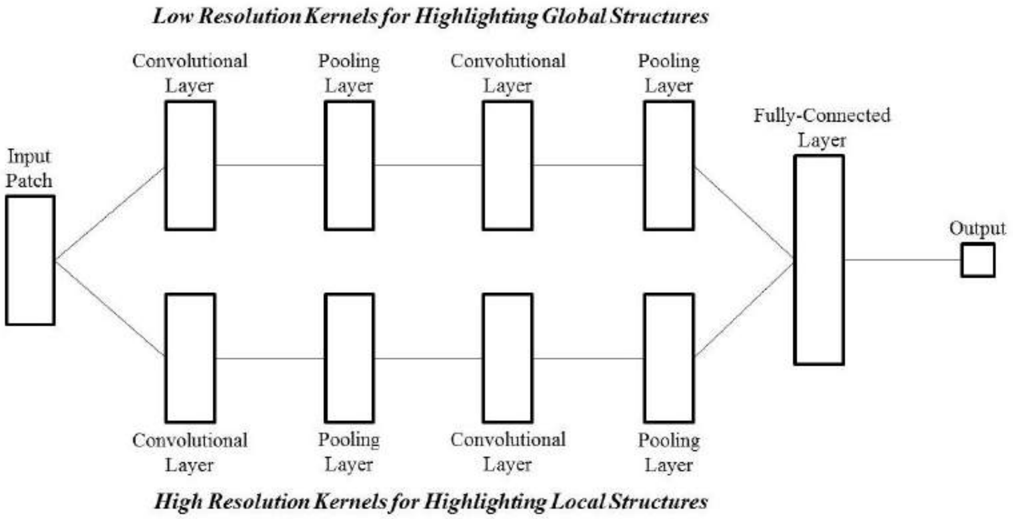


Figure 6.2: Example of potential CNN implementation, capable of learning both local and global features.

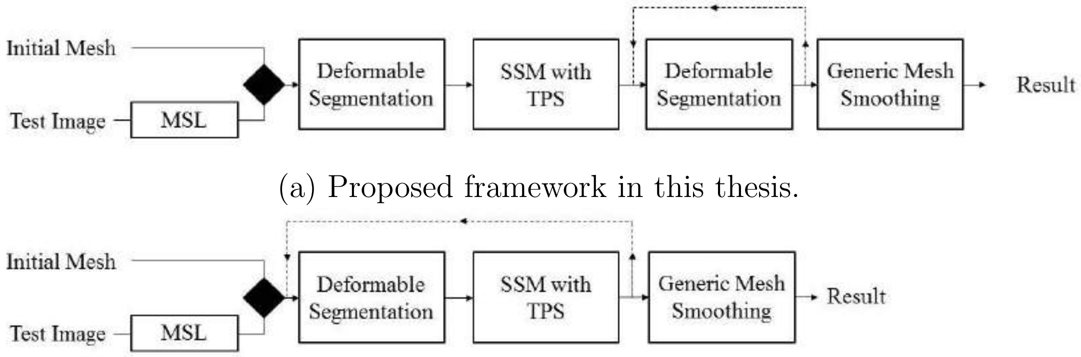
However, using a similar CNN approach for boundary detection in aortic root segmentation may not be suitable. CNNs require a local patch that is large enough to be able to convolve a kernel over, as this process yields a feature map smaller than the original patch due to the nature of convolution. However, it is only possible to extract a very small local patch for aortic boundary detection near the LVOT, due to its obscured appearance as a result of blood flow. As a result, multiple convolutional operations cannot be performed, and so sufficiently abstract filters may not be learned.

6.3.4 Decaying Shape Constraints

In its current state, the implementation of the proposed framework for aortic root segmentation is in sequential fashion. The *initial* segmentation can be thought of as the first deformable segmentation component followed by applying the SSM constraints. This is followed by an iterative process of the second deformable segmentation component, without applying shape constraints. During this iterative process the boundary detector search length is decreasing on every iteration, and so follows the following relationship;

$$s_{BD} \propto c_s \frac{1}{i} \tag{6.1}$$

where s_{BD} is the boundary detector search length, i is the iteration, and c_s is a constant. Figure 6.3 (a) shows a reminder of this proposed framework from Chapter 4.



(a) Proposed framework in this thesis.

(b) Proposed framework with decaying SSM weight.

Figure 6.3: Comparison study pipelines.

However, this can also be drawn in an ASM-like fashion (Figure 6.3 (b)), given the binary condition below;

$$w_{SSM}(i) = \begin{cases} 1, i = 1 \\ 0, i > 1 \end{cases} \tag{6.2}$$

where w_{SSM} is the weight given to the prior shape constraints. In future work it would be interesting to adjust the SSM so that it decays with each iteration, following the below relationship;

$$w_{SSM} \propto c_w \frac{1}{i} \tag{6.3}$$

where c_w is a constant.

It is thought that such an implementation might produce smoother mesh surfaces while still maintaining sufficient model flexibility. However, inevitably this would require many more iterations to converge, resulting in a longer testing time.

Furthermore the weighting parameter associated with the SSM constraints (w_{SSM}) is not easily determined, requiring trial-and-error for the best formulation. However, this implementation would provide an intermediate step between traditional ASM [53], and the proposed implementation presented in this thesis, in terms of flexibility and shape regularisation. It is envisaged that such a comparison would prove very useful for even further development.

6.3.5 Application

Furthermore, after accurately delineating the aortic root, future work can consist of segmenting the remaining vessel-like aortic structures. This includes the ascending and descending aorta, as well as the aortic arch. Accurate segmentation of these areas would provide completeness for TAVI analysis (Chapter 1). By providing full segmentation and structure measurements for these additional regions, clinicians may not only accurately plan the position and size of the prosthetic valve, but also its route to the target region. It is proposed that the aortic root be segmented using the approach presented in Chapter 4, and by using this knowledge the remaining aorta be segmented using the approach presented in Chapter 5.

Another application of such complete segmentation would be for planning pulmonary valve replacement (PVR). The pulmonary valve separates the right ventricle from the pulmonary artery, which carries oxygen-poor blood to the lungs for oxygenisation [99]. Pulmonary valve stenosis is a narrowing of the valve which causes decreased blood flow to the lungs, causing chest pains, fatigue and even sudden death. Like TAVI, percutaneous pulmonary valve implantation (PPVI) [71, 133, 165] is an operation to insert a bioprosthetic valve with a transcatheter. Commonly the catheter is passed through the pulmonary vein from the groin all the way to the valve, before depositing the prosthetic valve in place. This non-invasive procedure is extremely popular with clinicians, and medical imaging is paramount for its success. It is assumed that similar segmentation challenges would appear to aortic valve segmentation. The region between the valve and the right ventricle, called the right ventricular output tract (RVOT) would be extremely noisy due to inconsistent blood contrasts, however as demonstrated in

Chapter 4 this can be dealt with. Such segmentation can help clinicians with pulmonary valve measurement and operative planning in a similar way to aortic valve segmentation.

Finally, it is believed that the proposed deformable modelling framework is suitable for segmenting other large anatomical structures. Zheng et al. [285] and Grbic et al. [100] have demonstrated that explicit deformable modelling is capable of segmenting numerous heart structures such as the ventricles, atriums and valves, and Ling et al. [159] has demonstrated its use for liver segmentation. Given sufficient training images and example meshes, it is thought that multilayer NNs are capable of identifying features for accurate pose estimation and boundary detection in such cases, and that the presented mesh regularisation technique can smooth such surfaces. Furthermore it is thought that using NNs and automated SSMs can significantly increase the preparation time of segmenting such structures compared to this previous work [100, 159, 285, 288]. Chapters 4 and 5 have also shown that the framework is capable of segmenting structures in multiple imaging modalities. Interesting work may also be carried out by using the proposed framework to segment other organs in modalities such as MRI and PET images.

Appendix A

Evaluation Metrics

Euclidean Distance

Given two 3D coordinates p^A and p^B containing (x, y, z) dimensions, the Euclidean distance is computed as follows;

$$d(p^A, p^B) = \sqrt{(p_x^A - p_x^B)^2 + (p_y^A - p_y^B)^2 + (p_z^A - p_z^B)^2} \tag{A.1}$$

Where p_x^A is the x coordinate of p^A, and p_x^B is the x coordinate of p^B.

Point-to-Mesh Distance

The point-to-mesh distance provides an average error measure between two sets of point coordinates, A and B, and is computed as follows;

$$PMD(A, B) = \frac{1}{n} \sum_{i=1}^{N} d(p_i^A, p_i^B) \tag{A.2}$$

where $d(p_i^A, p_i^B)$ is the Euclidean distance between point i in set A, and it's nearest neighbour in set B. n is the number of points in set A. In terms of it's application to segmentation evaluation, A is often the ground truth coordinates while B is the segmentation result. However, in many cases the number of coordinates in A and B are not equal, therefore Equation A.2 can be biased towards one of the point sets. It is therefore often necessary to calculate the *symmetrical* point-to-mesh distance instead.

$$\overline{PMD}(A,B) = \frac{1}{n+m}(\sum_{i=1}^{n} d(p_i^A, p_i^B) + \sum_{i=1}^{m} d(p_i^B, p_i^A)) \tag{A.3}$$

Where $d(p_i^B, p_i^A)$ is the Euclidean distance between point i in set B, and it's nearest neighbour in set A, and m is the number of points in B.

Hausdorff Distance

Hausdorff distance provides a metric to quantify how far away the furthest point in A is from any point in B [42], and is calculated as follows;

$$HD(A,B) = \max_{i=1}^{n} |d(p_i^A, p_i^B)| \tag{A.4}$$

where $d(p_i^A, p_i^B)$ is the Euclidean distance between point i in set A, and it's nearest neighbour in set B, and n is the number of points in A. The symmetrical Hausdorff distance also considers the distance of the furthest point in B is from any point in A to give a more accurate measure.

$$\overline{HD}(A,B) = \max(\max_{i=1}^{n} |d(p_i^A, p_i^B)|, \max_{i=1}^{m} |d(p_i^B, p_i^A)|) \tag{A.5}$$

Here, $d(p_i^B, p_i^A)$ is the Euclidean distance between point i in set B, and it's nearest neighbour in set A, and m is the number of points in B.

Area Overlap

Area overlap is a ratio quantifying the amount of overlap between two sets. In terms of image segmentation, a positive sample may be a foreground pixel and a negative sample may be background. The area overlap is therefore the ratio between the segmented foreground pixels and the ground truth foreground pixels.

$$AO = \frac{TP}{TP + FP + FN} \times 100 \tag{A.6}$$

Where TP is the true-positive count (i.e. correctly classified foreground pixels), FP is the false-positive count (incorrectly classified background pixels), and FN is the false-positive count (incorrectly classified foreground pixels).

Specificity

Specificity is the true negative rate in binary classification. The measure quantifies the proportion of correctly identified negative samples as a percentage.

$$Specificity = \frac{TN}{N} \times 100 \tag{A.7}$$

$$Specificity = \frac{TN}{TN + FN} \times 100 \tag{A.8}$$

Where N is the total number of samples classed as negative, TN is the true-negative count, and FN is the false-negative count.

Sensitivity

Sensitivity is the true positive rate in binary classification. The measure quantifies the proportion of correctly identified positive samples as a percentage.

$$Sensitivity = \frac{TP}{P} \times 100 \tag{A.9}$$

$$Sensitivity = \frac{TP}{TP + FP} \times 100 \tag{A.10}$$

Where P is the total number of samples classed as positive, TP is the true-positive count, and FP is the false-positive count.

Accuracy

Accuracy is a measure that quantifies the number of correctly classified samples in a binary classification problem, as a percentage. It is the number of correctly classifications divided by the number of classifications attempted.

$$Accuracy = \frac{TN + TP}{N + P} \times 100 \tag{A.11}$$

$$Accuracy = \frac{TN + TP}{TN + FN + TP + FP} \times 100 \tag{A.12}$$

Where N is the total number of samples classified as negative, and P is the total number of samples classified as positive. TN is the true negative count, FN is the false negative count, TP is the true positive count, and FP is the false positive count.

Appendix B

Symbols

The following lists describe important symbols used for describing the proposed methods in Chapters 3, 4 and 5, and Appendix A.

Chapter 3: Dense Mesh Correspondence

$\mathbf{A}$: Affine matrix containing rotation and scale parameters.

$\mathbf{b}$: Affine translation vector.

$\mathbf{E_r}$: Smoothness cost.

$\mathbf{E_s}$: Mesh-based similarity metric.

$\mathbf{H}$: Number of mesh resolutions.

$\mathbf{I_s}$: Source image.

$\mathbf{I_t}$: Target image.

$\mathbf{M_s}(\mathbf{V_s}, \mathbf{E_s}, \mathbf{F_s})$: Source mesh with vertices V_s, edges E_s, and faces F_s.

$\mathbf{M_s'}$: Transformed source mesh after one local transformations.

$\mathbf{M_s''}$: Transformed source mesh after two local transformations.

$\mathbf{M_t}(\mathbf{V_t}, \mathbf{E_t}, \mathbf{F_t})$: Target mesh with vertices V_t, edges E_t, and faces F_t.

$\mathbf{P_s}$: Subset of landmark vertices in V_s.

$\mathbf{P_t}$: Subset of landmark vertices in V_t.

$\mathbf{Q_s}$: Subset of vertices in V_s that are correspondent with V_t.

$\mathbf{Q_s'}$: Subset of vertices in V_s' that are correspondent with V_t.

$\mathbf{Q_s''}$: Subset of vertices in V_s'' that are correspondent with V_t.

$\mathbf{T}$: Transformation.

$\mathbf{T_g}$: Global transformation.

$\mathbf{T_l}$: Local transformation.

$\mathbf{T_{l_1}}$: 1st local transformation.

$\mathbf{T_{l_2}}$: 2nd local transformation.

$\mathbf{x}$: Mesh coordinates.

$\delta_\mathbf{0}$: Initial control point spacing.

$\delta_\mathbf{h}$: ϕ spacing at resolution h.

φ: Regularisation term.

$\phi_\mathbf{ijk}$: FFD control points.

Chapter 4: Aortic Root Segmentation

$\mathbf{a_j^{k+1}}$: NN activation of node j in layer $k+1$.

$\mathbf{B_l}$,$\mathbf{B_m}$ and $\mathbf{B_n}$: Basis functions of the B-spline.

$\mathbf{b}$: Shape parameters.

$\mathbf{b_j^{k+1}}$: Bias of node j in layer $k+1$.

$\mathbf{C_a}$: Position space classifier.

$\mathbf{C_{ab}}$: Position-orientation space classifier.

$\mathbf{C_{abc}}$: Position-orientation-scale space classifier.

$\mathbf{E_r}$: Smoothness cost.

$\mathbf{E_s}$: Mesh-based similarity metric.

$\mathbf{f}$: Sigmoid activation function.

$\mathbf{H}$: Number of mesh resolutions.

$\mathbf{m}$: Shape variability parameter.

$\mathbf{M_s(V_s, E_s, F_s)}$: Source mesh with vertices V_s, edges E_s, and faces F_s.

$\mathbf{M'_s}$: Transformed source mesh after one local transformations.

$\mathbf{M_t(V_t, E_t, F_t)}$: Target mesh with vertices V_t, edges E_t, and faces F_t.

$\mathbf{n}$: Number of vertices in V.

$\mathbf{P}$: Eigenvectors.

$\mathbf{P_k}$: Pose parameter hypothesis.

$\mathbf{P_s}$: Subset of landmark vertices in V_s.

$\mathbf{P'_s}$: Subset of vertices in V'_s that are correspondent with P_t.

$\mathbf{P_t}$: Subset of landmark vertices in V_t.

$\mathbf{S_k}$: Parameter search step.

$\mathbf{S_x}$: Local scale in x dimension.

$\overline{\mathbf{S^t_x}}$: Mean ground truth local scale in x dimension.

$\mathbf{S_y}$: Local scale in y dimension.

$\overline{\mathbf{S^t_y}}$: Mean ground truth local scale in y dimension.

$\mathbf{S_z}$: Local scale in z dimension.

$\overline{\mathbf{S^t_z}}$: Mean ground truth local scale in z dimension.

$\mathbf{T}$: Transformation.

$\mathbf{T_g}$: Global transformation.

$\mathbf{T_l}$: Local transformation.

$\mathbf{V}$: Original mesh vertices.

$\mathbf{V'}$: Desired mesh vertices after boundary detection.

$\mathbf{w^k_{ji}}$: Weight connection between node i in layer k and node j in layer $k+1$.

$\mathbf{x}$: x coordinate.

$\mathbf{x_t}$: Ground truth x coordinate.

$\mathbf{y}$: x coordinate.

$\mathbf{y_t}$: Ground truth y coordinate.

$\mathbf{z}$: x coordinate.

$\mathbf{z_t}$: Ground truth z coordinate.

$\delta_\mathbf{0}$: Initial control point spacing.

$\delta_\mathbf{h}$: ϕ_{ijk} spacing at resolution h.

θ: 3rd Euler angle.

$\theta_\mathbf{t}$: 3rd Euler angle ground truth.

λ: Eigenvalues.

$\mu^\mathbf{k}_\mathbf{i}$: Input of node i in layer k.

$\mathbf{\Pi}$: Optimal pose hypothesis.

$\mathbf{\Pi_a}$: High scoring position hypotheses in Ψ_a.

$\mathbf{\Pi_{ab}}$: High scoring position-orientation hypotheses in Ψ_{ab}.

$\mathbf{\Pi_{abc}}$: High scoring position-orientation-scale hypotheses in Ψ_{abc}.

$\mathbf{\Upsilon}$: Matrix of S arrays υ.

υ: 1D array of 3D landmark coordinates.

$\bar{\upsilon}$: Mean 1D array 3D landmark coordinates.

ϕ: 2nd Euler angle.

$\phi_\mathbf{ijk}$: FFD control points.

$\phi_\mathbf{t}$: 2nd Euler angle ground truth.

φ: Regularisation term.

$\mathbf{\Psi}$: High dimensional search space.

$\mathbf{\Psi_a}$: Position search space.

$\mathbf{\Psi_{ab}}$: Position-orientation search space.

$\mathbf{\Psi_{abc}}$: Position-orientation-scale space.

ω: 1st Euler angle.

$\omega_{\mathbf{t}}$: 1st Euler angle ground truth.

Chapter 5: Lymphatic Vessel Segmentation

$\mathbf{a}$,$\mathbf{b}$,$\mathbf{c}$,$\mathbf{d}$,$\mathbf{e}$ and $\mathbf{f}$: Ellipse parameters.

$\mathbf{C}$: Original contour points.

$\mathbf{C'}$: New contour points after boundary detection.

$\mathbf{c_i^{cart}}$: 2D contour of slice i in cartesian coordinates.

$\mathbf{c_i^{polar}}$: 2D contour of slice i in polar coordinates.

$\mathbf{f_{in}}$: Haar filter output for inner vessel wall.

$\mathbf{f_{out}}$: Haar filter output for outer vessel wall.

$\mathbf{l}$: Column search leeway.

$\mathbf{N}$: Number of pixels in one Haar rectangle.

$\mathbf{s_i^{cart}}$: Image slice i in cartesian coordinates.

$\mathbf{s_i^{polar}}$: Image slice i in polar coordinates.

$\mathbf{T}$: Transformation.

$\mathbf{t}$: 2D outlier threshold.

$\mathbf{V}$: Original mesh vertices.

$\mathbf{V'}$: Desired mesh vertices after

$\mathbf{x}$: x coordinate.

$\mathbf{y}$: y coordinate.

δ: Control point spacing.

μ: Pixel intensity.

$\phi_{\mathbf{ijk}}$: FFD control points.

Chapter 6: Conclusions and Future work

$\mathbf{c_s}$: Constant associated with boundary detector search length.

$\mathbf{c_w}$: Constant associated with SSM weight.

$\mathbf{i}$: Iteration.

$\mathbf{s_{BD}}$: BD search length.

$\mathbf{w_{SSM}}$: SSM weight.

Appendix A: Evaluation Metrics

A: Set of 3D coordinates.

AO: Area overlap.

B: Set of 3D coordinates.

d: Euclidean distance.

FN: Number of false negatives.

FP: Number of false positives.

HD: Hausdorff distance.

$\overline{\mathbf{HD}}$: Symmetrical Hausdorff distance.

m: Number of coordinates in B.

N: Number of samples classified as negative.

n: Number of coordinates in A.

P: Number of samples classified as positive.

PMD: Point to mesh error.

$\overline{\mathbf{PMD}}$: Symmetrical point to mesh error.

$\mathbf{p^A}$: 3D coordinate in A.

$\mathbf{p^B}$: 3D coordinate in B.

TN: Number of true negatives.

TP: Number of true positives.

Appendix C

Acronyms

AAM: Active Appearance Model

AdaBoost: Adaptive Boosting

AO: Area Overlap

ASM: Active Shape Model

BD: Boundary Detector

CART: Classification and Regression Trees

CNN: Convolutional Neural Network

CNN-BD: Convolutional Neural Network Boundary Detector

CNN-MSL: Convolutional Neural Network Marginal Space Learning

CT: Computed Tomography

EM: Expectation Maximisation

FFD: Free-Form-Deformation

HD: Hausdorff Distance

$\overline{\mathbf{HD}}$: Symmetrical Hausdorff Distance

HOG: Histogram of Gradient

LDA: Linear Discriminant Analysis

LVOT: Left Ventricular Output Tract

MDL: Minimum Description Length

MRI: Magnetic Resonance Imaging

MSDL: Marginal Space Deep Learning

MSL: Marginal Space Learning

NMR: Nuclear Magnetic Resonance

NN: Neural Network

NN-BD: Neural Network Boundary Detector

NN-MSL: Neural Network Marginal-Space-Learning

PCA: Principal Component Analysis

PET: Positron Emission Tomography

PMD: Point to Mesh Distance

$\overline{\mathbf{PMD}}$: Symmetrical Point to Mesh Distance

PPVI: Percutaneous Pulmonary Valve Implantation

PVR: Pulmonary Valve Replacement

RF: Random Forest

RF-BD: Random Forest Boundary Detector

RF-MSL: Random Forest Marginal Space Learning

ROC: Receiver Operating Characteristic

ROI: Region of Interest

RVOT: Right Ventricular Output Tract

SAVI: Surgical Aortic Valve Implantation

SAVR: Surgical Aortic Valve Replacement

SHOG: Spherical Histogram of Gradient

SPECT: Single Photon Emission Computed Tomography

SSD: Sum of Squared Distance

SSM: Statistical Shape Model

SVM: Support Vector Machine

TAVI: Transcatheter Aortic Valve Implantation

TAVR: Transcatheter Aortic Valve Replacement

TPS: Thin Plate Spline

Appendix D

Bibliography

[1] H. Abdi and L.J. Williams. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4):433–459, 2010.

[2] M. Abdoli, R. Dierckx, and H. Zaidi. Contourlet-based active contour model for pet image segmentation. *Medical physics*, 40(8):082507, 2013.

[3] R. Abu-Gharbieh, G. Hamarneh, and T. Gustavsson. Review-active shape models-part ii: Image search and classification. In *In Proc. Swedish Symposium on Image Analysis*. Citeseer, 1998.

[4] C.C. Aggarwal. *Data mining: The textbook*. Springer, 2015.

[5] A.M. Aisen, W. Martel, E.M. Braunstein, K.I. McMillin, W.A. Phillips, and T.F. Kling. Mri and ct evaluation of primary bone and soft-tissue tumors. *American journal of Roentgenology*, 146(4):749–756, 1986.

[6] A.A. Amini, S. Tehrani, and T.E. Weymouth. Using dynamic programming for minimizing the energy of active contours in the presence of hard constraints. In *Computer Vision., Second International Conference on*, pages 95–99. IEEE, 1988.

[7] AA. Amini, T.E. Weymouth, and R.C. Jain. Using dynamic programming for solving variational problems in vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(9):855–867, 1990.

[8] N.C. Andreasen. *Brain imaging: Applications in psychiatry*, volume 2. American Psychiatric Pub, 1989.

[9] O.C. Avila-Montes, U. Kukure, and I.A. Kakadiaris. Aorta segmentation in non-contrast cardiac ct images using an entropy-based cost function. In *SPIE Medical Imaging*, pages 76233J–76233J. International Society for Optics and Photonics, 2010.

[10] A. Baker, H. Kim, J.L. Semple, D. Dumont, M. Shoichet, D. Tobbia, and M. Johnston. Experimental assessment of pro-lymphangiogenic growth factors in the treatment of post-surgical lymphedema following lymphadenectomy. *Breast Cancer Research*, 12(5):R70, 2010.

[11] A. Balaji. Neuroradiology cases. `http://www.neuroradiologycases.com/2013/09/measles-encephalitis-mri.html`. Online; accessed 15-05-2016.

[12] P. Baluk and D.M. McDonald. Markers for microscopic imaging of lymphangiogenesis and angiogenesis. *Annals of the New York Academy of Sciences*, 1131(1):1–12, 2008.

[13] I.N. Bankman and S. Morcovescu. Handbook of medical imaging. processing and analysis. *Medical Physics*, 29(1):107–107, 2002.

[14] S. Bauer, R. Wiest, L.P. Nolte, and M. Reyes. A survey of mri-based medical image analysis for brain tumor studies. *Physics in medicine and biology*, 58(13):R97, 2013.

[15] P.J. Besl and N.D. McKay. Method for registration of 3-d shapes. In *Robotics-DL tentative*, pages 586–606. International Society for Optics and Photonics, 1992.

[16] A. Blake and M. Isard. *Active contours: the application of techniques from graphics, vision, control theory and statistics to visual tracking of shapes in motion.* Springer Science & Business Media, 2012.

[17] G.S. Bloomfield, L.D. Gillam, R.T. Hahn, S. Kapadia, J. Leipsic, S. Lerakis, M. Tuzcu, and P.S. Douglas. A practical guide to multimodality imaging

of transcatheter aortic valve replacement. *JACC: Cardiovascular Imaging*, 5(4):441–455, 2012.

[18] L. Bo, X. Ren, and D. Fox. Unsupervised feature learning for rgb-d based object recognition. In *Experimental Robotics*, pages 387–402. Springer, 2013.

[19] J. Bogaert. *Cardiac function*. Springer, 2011.

[20] T. Bolkart and S. Wuhrer. A groupwise multilinear correspondence optimization for 3d faces. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3604–3612, 2015.

[21] F. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE T-PAMI*, 11(6):567–585, 1989.

[22] J. Borovicka. Circle detection using hough transforms documentation. *COMS30121-Image Processing and Computer Vision*, 2003.

[23] A. Bosch, A. Zisserman, and X. Munoz. Image classification using random forests and ferns. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007.

[24] Y. Boykov and G. Funka-Lea. Graph cuts and efficient nd image segmentation. *International journal of computer vision*, 70(2):109–131, 2006.

[25] Y. Boykov and M.P. Jolly. Interactive organ segmentation using graph cuts. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2000*, pages 276–286. Springer, 2000.

[26] Y. Boykov and M.P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 105–112. IEEE, 2001.

[27] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. In *Energy minimization methods in computer vision and pattern recognition*, pages 359–374. Springer, 2001.

[28] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother. Learning 6d object pose estimation using 3d object coordinates. In *European Conference on Computer Vision*, pages 536–551. Springer, 2014.

[29] L. Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[30] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[31] L. Breiman and A. Cutler. A deterministic algorithm for global optimization. *Mathematical Programming*, 58(1-3):179–199, 1993.

[32] A.D. Brett and C.J. Taylor. A method of automated landmark generation for automated 3d pdm construction. *Image and Vision Computing*, 18(9):739–748, 2000.

[33] A. Broggi, E. Cardarelli, S. Cattani, P. Medici, and M. Sabbatelli. Vehicle detection for autonomous parking using a soft-cascade adaboost classifier. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pages 912–917. IEEE, 2014.

[34] M.J. Budoff and J.S. Shinbane. *Handbook of cardiovascular CT: Essentials for clinical practice.* Springer Science & Business Media, 2008.

[35] A.J. Bulpitt and N.D. Efford. An efficient 3d deformable model with a self-optimising topology. In *BMVC*, pages 1–10, 1995.

[36] A.J. Bulpitt and N.D. Efford. An efficient 3d deformable model with a self-optimising mesh. *Image and Vision Computing*, 14(8):573–580, 1996.

[37] J. Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):679–698, 1986.

[38] V. Caselles, F. Catté, and F. Coll, T.and Dibos. A geometric model for active contours in image processing. *Numerische mathematik*, 66(1):1–31, 1993.

[39] P.J. Caspers, G.W. Lucassen, and G.J. Puppels. Combined in vivo confocal raman spectroscopy and confocal microscopy of human skin. *Biophysical journal*, 85(1):572–580, 2003.

[40] H.D. Cavanagh, J.V. Jester, J. Essepian, W. Shields, and M.A. Lemp. Confocal microscopy of the living eye. *Eye & Contact Lens*, 16(1):65–73, 1990.

[41] Heart Matters Medical Centre. Transcatheter aortic valve implantation a.k.a tavi. `http://www.heartmatters.com.sg/transcatheter-aortic-valve-implantation-a-k-a-tavi/#.V9VwPNP2bDc`. Online; accessed 11-09-2016.

[42] V. Chalana and Y. Kim. A methodology for evaluation of boundary detection algorithms on medical images. *Medical Imaging, IEEE Transactions on*, 16(5):642–652, 1997.

[43] A. Charnes, E.L. Frome, and P.L. Yu. The equivalence of generalized least squares and maximum likelihood estimates in the exponential family. *Journal of the American Statistical Association*, 71(353):169–171, 1976.

[44] N.J. Chatap and A.K. Shrivastava. A survey on various classification techniques for medical image data. *International Journal of Computer Applications*, 97(15), 2014.

[45] P.C. Chen and T. Pavlidis. Segmentation by texture using a co-occurrence matrix and a split-and-merge algorithm. *Computer graphics and image processing*, 10(2):172–182, 1979.

[46] D. Chetverikov, D. Svirko, D. Stepanov, and P. Krsek. The trimmed iterative closest point algorithm. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 3, pages 545–548. IEEE, 2002.

[47] A. Cheung and K.M. Lichtenstein. Illustrated techniques for transapical aortic valve implantation. *Annals of Cardiothoracic Surgery*, 1(2):231, 2012.

[48] A. Cheung and R. Ree. Transcatheter aortic valve replacement. *Anesthesiology clinics*, 26(3):465–479, 2008.

[49] L. Chunming. Active contours, level sets, and image segmentation. `http://slideplayer.com/slide/3431698/`. Online; accessed 14-05-2016.

[50] L.D. Cohen. On active contour models and balloons. *CVGIP: Image understanding*, 53(2):211–218, 1991.

[51] L.D. Cohen and I. Cohen. Finite-element methods for active contour models and balloons for 2-d and 3-d images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 15(11):1131–1147, 1993.

[52] T. Cootes, ER Baldock, and J Graham. An introduction to active shape models. *Image processing and analysis*, pages 223–248, 2000.

[53] T. Cootes, D.H. Cooper, and J. Graham. Active shape models - their training and application. *CVIU*, 61(1):38–59, 1995.

[54] T. Cootes, G. Edwards, and C.J. Taylor. Active appearance models. In *Computer Vision—ECCV'98*, pages 484–498. Springer, 1998.

[55] T. Cootes, G. Edwards, and C.J. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (6):681–685, 2001.

[56] T. Cootes, A. Hill, C.J. Taylor, and J. Haslam. Use of active shape models for locating structures in medical images. *Image and vision computing*, 12(6):355–365, 1994.

[57] T. Cootes and C.J. Taylor. Statistical models of appearance for medical image analysis and computer vision. In *Medical Imaging 2001*, pages 236–248. International Society for Optics and Photonics, 2001.

[58] T. Cootes, G. Wheeler, K. Walker, and C.J. Taylor. View-based active appearance models. *Image and vision computing*, 20(9):657–664, 2002.

[59] D. Cremers, M. Rousson, and R. Deriche. A review of statistical approaches to level set segmentation: integrating color, texture, motion and shape. *International journal of computer vision*, 72(2):195–215, 2007.

[60] A. Cribier, H. Eltchaninoff, A. Bash, N. Borenstein, C. Tron, F. Bauer, G. Derumeaux, F. Anselme, F. Laborde, and M.B. Leon. Percutaneous transcatheter implantation of an aortic valve prosthesis for calcific aortic stenosis first human case description. *Circulation*, 106(24):3006–3008, 2002.

[61] A. Criminisi, D. Robertson, E. Konukoglu, J. Shotton, S. Pathak, S. White, and K. Siddiqui. Regression forests for efficient anatomy detection and localization in computed tomography scans. *Medical image analysis*, 17(8):1293–1303, 2013.

[62] R.H. Davies, C.J. Twining, T.F. Cootes, J.C. Waterton, and C.J Taylor. 3d statistical shape models using direct optimisation of description length. In *Computer Vision—ECCV 2002*, pages 3–20. Springer, 2002.

[63] R.H. Davies, C.J. Twining, T.F. Cootes, J.C. Waterton, and C.J. Taylor. A minimum description length approach to statistical shape modeling. *Medical Imaging, IEEE Transactions on*, 21(5):525–537, 2002.

[64] M. de Bruijne, B. van Ginneken, M.A. Viergever, and W.J. Niessen. Adapting active shape models for 3d segmentation of tubular structures in medical images. In *Information Processing in Medical Imaging*, pages 136–147. Springer, 2003.

[65] H. Delingette. Simplex meshes: a general representation for 3d shape reconstruction. In *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on*, pages 856–859. IEEE, 1994.

[66] E.W. Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.

[67] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, pages 647–655, 2014.

[68] J.S. Duncan and N. Ayache. Medical image analysis: Progress over two decades and the challenges ahead. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(1):85–106, 2000.

[69] J.P. Earls, V.B. Ho, T.K. Foo, E. Castillo, and S.D. Flamm. Cardiac mri: recent progress and continued challenges. *Journal of magnetic resonance imaging*, 16(2):111–127, 2002.

[70] O. Ecabert, J. Peters, H. Schramm, C. Lorenz, J. Von Berg, M.J. Walker, M. Vembar, M.E. Olszewski, K. Subramanyan, G. Lavi, et al. Automatic model-based segmentation of the heart in ct images. *Medical Imaging, IEEE Transactions on*, 27(9):1189–1201, 2008.

[71] A. Eicken, P. Ewert, A. Hager, B. Peters, S. Fratz, T. Kuehne, R. Busch, J. Hess, and F. Berger. Percutaneous pulmonary valve implantation: two-centre experience with more than 100 patients. *European heart journal*, 32(10):1260–1265, 2011.

[72] A. Elattar, M, E.M. Wiegerinck, R.N. Planken, E. vanbavel, H.C. van Assen, and J. Baan Jr. Automatic segmentation of the aortic root in ct angiography of candidate patients for transcatheter aortic valve implantation. *Med. Biol. Eng. Comput.*, 52:611–618, 2014.

[73] A. Ericsson and K. Åström. Minimizing the description length using steepest descent. In *BMVC*, pages 1–10, 2003.

[74] L. Espona, M.J. Carreira, M.G. Penedo, and M. Ortega. Retinal vessel tree segmentation using a deformable contour model. In *19th ICPR*, pages 2128–2131, 2008.

[75] E. Essa, X. Xie, and J.L. Jones. Minimum s-excess graph for segmenting and tracking multiple borders with hmm. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015*, pages 28–35. Springer, 2015.

[76] B. Evelian. X-ray computed tomography (ct scans). `http://quakerecnankai.blogspot.co.uk/2014/12/`

x-ray-computed-tomography-ct-scans.html. Online; accessed 14-05-2016.

[77] D. Feng. *Segmentation of soft tissues in medical images.* PhD thesis, National University of Singapore, 2005.

[78] M.A.T. Figueiredo, J. Leitão, and A.K. Jain. Adaptive b-splines and boundary estimation. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 724–730. IEEE, 1997.

[79] M. Flower. *Webb's physics of medical imaging.* CRC Press, 2012.

[80] G.D. Forney Jr. The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, 1973.

[81] A.F. Frangi, W.J. Niessen, D. Rueckert, and J.A. Schnabel. Automatic 3d asm construction via atlas-based landmarking and volumetric elastic registration. In *Information Processing in Medical Imaging*, pages 78–91. Springer, 2001.

[82] A.F. Frangi, D. Rueckert, J.A. Schnabel, and W.J. Niessen. Automatic construction of multiple-object three-dimensional statistical shape models: Application to cardiac modeling. *IEEE T-MI*, 21(9):1151–1166, 2002.

[83] W.T. Freeman and E.H. Adelson. The design and use of steerable filters. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 9:891–906, 1991.

[84] Y. Freund, R. Schapire, and N. Abe. A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612, 1999.

[85] J. Friedman, T. Hastie, R. Tibshirani, et al. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 28(2):337–407, 2000.

[86] W.H. Gaasch and R.S. D'Agostino. Transcatheter aortic valve implantation: the transfemoral versus the transapical approach. *Annals of Cardiothoracic Surgery*, 1(2):200, 2012.

[87] J. Gall and V. Lempitsky. Class-specific hough forests for object detection. In *Decision forests for computer vision and medical image analysis*, pages 143–157. Springer, 2013.

[88] Y. Gan, Z. Xia, J. Xiong, Q. Zhao, Y. Hu, and J. Zhang. Toward accurate tooth segmentation from computed tomography images using a hybrid level set model. *Medical physics*, 42(1):14–27, 2015.

[89] A.P. Ganatra and Y.P. Kosta. Comprehensive evolution and evaluation of boosting. *International Journal of Computer Theory and Engineering*, 2(6):931, 2010.

[90] Q. Ge, C. Li, W. Shao, and H. Li. A hybrid active contour model with structured feature for image segmentation. *Signal Processing*, 108:147–158, 2015.

[91] P. Généreux, S.J. Head, D.A. Wood, S.K. Kodali, M.R. Williams, J.M. Paradis, M. Spaziano, A.P. Kappetein, J.G. Webb, A. Cribier, et al. Transcatheter aortic valve implantation 10-year anniversary: review of current evidence and clinical implications. *European heart journal*, 33(19):2388–2398, 2012.

[92] B. Georgescu, X.S. Zhou, D. Comaniciu, and A. Gupta. Database-guided segmentation of anatomical structures with complex appearance. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 429–436. IEEE, 2005.

[93] M. Gessat, T. Frauenfelder, L. Altwegg, J. Grunenfelder, and V. Falk. Transcatheter aortic valve implantation: Role of imaging. *Aswan Heart Centre Science and Practice Series*, 1(3), 2011.

[94] F.C. Ghesu, B. Georgescu, Y. Zheng, J. Hornegger, and D. Comaniciu. Marginal space deep learning: Efficient architecture for detection in volumetric image data. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015*, pages 710–718. Springer, 2015.

[95] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[96] S.T. Gollmer, M. Kirschner, T.M. Buzug, and S. Wesarg. Using image segmentation for evaluating 3d statistical shape models built with groupwise correspondence optimization. *Computer Vision and Image Understanding*, 125:283–303, 2014.

[97] J.C. Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, 1975.

[98] I.S. Grant and W.R. Phillips. *Electromagnetism.* John Wiley & Sons, 2013.

[99] H. Gray. *Gray's Anatomy: With original illustrations by Henry Carter.* Arcturus Publishing, 2009.

[100] S. Grbic, R. Ionasec, D. Vitanovski, I. Voigt, B. Georgescu, N. Navab, and D. Comaniciu. Complete valvular heart apparatus model from 4d cardiac ct. *Med. Image Anal.*, 16:1003–1014, 2012.

[101] D.M. Greig, B.T. Porteous, and A.H. Seheult. Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 271–279, 1989.

[102] T. Hara, N. Kosaka, and H. Kishi. Pet imaging of prostate cancer using carbon-11-choline. *The Journal of Nuclear Medicine*, 39(6):990, 1998.

[103] S. Haykin. A comprehensive foundation. *Neural Networks*, 2(2004), 2004.

[104] Unity Point Health. Transcatheter aortic valve replacement (tavr), 2016. Online; accessed 04-07-2016.

[105] Intermountain Healthcare. Aortic valve stenosis. `https://intermountainhealthcare.org/services/heart-care/conditions/aortic-valve-stenosis/`. Online; accessed 07-04-2016.

[106] heartlungdoc.com. Trans-catheter aortic valve replacement (tavr) surgery. `http://www.heartlungdoc.com/index.php/en/the-heart/trans-catheter/tavrc`. Online; accessed 11-09-2016.

[107] T. Heimann and H.P. Meinzer. Statistical shape models for 3d medical image segmentation: a review. *Medical image analysis*, 13(4):543–563, 2009.

[108] T. Heimann, H.P. Meinzer, and I. Wolf. A statistical deformable model for the segmentation of liver ct volumes. *3D Segmentation in the clinic: A grand challenge*, pages 161–166, 2007.

[109] T. Heimann, S. Münzing, H.P. Meinzer, and I. Wolf. A shape-guided deformable model with evolutionary algorithm initialization for 3d soft tissue segmentation. In *Information Processing in Medical Imaging*, pages 1–12. Springer, 2007.

[110] R. Helen and N. Kamaraj. Cad scheme to detect brain tumour in mr images using active contour models and tree classifiers. *Journal of Electrical Engineering and Technology*, 10(2):670–675, 2015.

[111] D.L.G. Hill, P.G. Batchelor, M. Holden, and D.J. Hawkes. Medical image registration. *Physics in medicine and biology*, 46(3):R1, 2001.

[112] S. Hinterstoisser, C. Cagniart, S. Ilic, P. Sturm, N. Navab, P. Fua, and V. Lepetit. Gradient response maps for real-time detection of textureless objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(5):876–888, 2012.

[113] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Asian conference on computer vision*, pages 548–562. Springer, 2012.

[114] C.M.A. Hoeks, J.O. Barentsz, T. Hambrock, D. Yakar, D.M. Somford, S.W.T.P.J. Heijmink, T.W.J. Scheenen, P.C. Vos, H. Huisman, I.M. van Oort, et al. Prostate cancer: multiparametric mr imaging for detection, localization, and staging. *Radiology*, 261(1):46–66, 2011.

[115] S. Holzer, J. Shotton, and P. Kohli. Learning to efficiently detect repeatable interest points in depth data. In *European Conference on Computer Vision*, pages 200–213. Springer, 2012.

[116] P.V.C. Hough. Method and means for recognizing complex patterns. Technical report, 1962.

[117] Y.L. Hu, W.J. Rogers, D.A. Coast, C.M. Kramer, and N. Reichek. Vessel boundary extraction based on a global and local deformable physical model with variable stiffness. *MRI*, 16:943–951, 1998.

[118] D.P. Huttenlocher, G.A. Klanderman, and W.J. Rucklidge. Comparing images using the hausdorff distance. *IEEE Transactions on pattern analysis and machine intelligence*, 15(9):850–863, 1993.

[119] Mater Imaging. Obstetric ultrasound. `http://www.materimaging.com.au/Obstetric.html`. Online; accessed 15-05-2016.

[120] R.I. Ionasec, I. Voigt, B. Georgescu, Y. Wang, H. Houle, F. Vega-Higuera, N. Navab, and D. Comaniciu. Patient-specific modeling and quantification of the aortic and mitral valves from 4-d cardiac ct and tee. *Medical Imaging, IEEE Transactions on*, 29(9):1636–1651, 2010.

[121] V. Jain and H. Zhang. Robust 3d shape correspondence in the spectral domain. In *Shape Modeling and Applications, 2006. SMI 2006. IEEE International Conference on*, pages 19–19. IEEE, 2006.

[122] J. Jiang, P. Trundle, and J. Ren. Medical image analysis with artificial neural networks. *Computerized Medical Imaging and Graphics*, 34(8):617–631, 2010.

[123] I. Jolliffe. *Principal component analysis*. Wiley Online Library, 2002.

[124] M. Jones and P. Viola. Fast multi-view face detection. *Mitsubishi Electric Research Lab TR-20003-96*, 3:14, 2003.

[125] C.M. Jude, N.B. Nayak, M.K. Patel, M. Deshmukh, and P. Batra. Pulmonary coccidioidomycosis: pictorial review of chest radiographic and ct findings. *RadioGraphics*, 34(4):912–925, 2014.

[126] F. Jung, M. Kirschner, and S. Wesarg. A generic approach to organ detection using 3d haar-like features. In *Bildverarbeitung für die Medizin*, pages 320–325. Springer, 2013.

[127] D. Kainmueller. Deformable meshes for automatic segmentation. In *Deformable Meshes for Medical Image Segmentation*, pages 25–49. Springer, 2015.

[128] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.

[129] M.R. Kaus, V. Pekar, C. Lorenz, R. Truyen, S. Lobregt, and J. Weese. Automated 3-d pdm construction from segmented images using deformable models. *Medical Imaging, IEEE Transactions on*, 22(8):1005–1013, 2003.

[130] M.R. Kaus, J. von Berg, J. Weese, W. Niessen, and V. Pekar. Automated segmentation of the left ventricle in cardiac mri. *Medical image analysis*, 8(3):245–254, 2004.

[131] W. Kehl, F. Milletari, F. Tombari, S. Ilic, and N. Navab. Deep learning of local rgb-d patches for 3d object detection and 6d pose estimation. *arXiv preprint arXiv:1607.06038*, 2016.

[132] B.M. Kelm, M. Wels, S. Zhou, S. Seifert, M. Suehling, Y. Zheng, and D. Comaniciu. Spine detection in ct and mr using iterated marginal space learning. *Medical image analysis*, 17(8):1283–1292, 2013.

[133] S. Khambadkone, L. Coats, A. Taylor, Y. Boudjemline, G. Derrick, V. Tsang, J. Cooper, V. Muthurangu, S.R. Hegde, R.S. Razavi, et al. Percutaneous pulmonary valve implantation in humans results in 59 consecutive patients. *Circulation*, 112(8):1189–1197, 2005.

[134] T.K. Kim, I. Budvytis, and R. Cipolla. Making a shallow network deep: Conversion of a boosting classifier into a decision tree by boolean optimisation. *International journal of computer vision*, 100(2):203–215, 2012.

[135] V.G. Kim, Y. Lipman, and T. Funkhouser. Blended intrinsic maps. In *ACM Transactions on Graphics (TOG)*, volume 30, page 79. ACM, 2011.

[136] C. Kirbas and F. Quek. A review of vessel extraction techniques and algorithms. *ACM Computing Surveys (CSUR)*, 36(2):81–121, 2004.

[137] T. Kohlberger, M. Sofka, J. Wetzl, J. Zhang, S.K. Zhou, N. Birkbeck, J. Declerck, and J. Kaftan. Method and system for multi-organ segmentation using learning-based segmentation and level set optimization, May 26 2015. US Patent 9,042,620.

[138] R. Korez, B. Ibragimov, B. Likar, F. Pernuš, and T. Vrtovec. Interpolation-based shape-constrained deformable model approach for segmentation of vertebrae from ct spine images. In *Recent Advances in Computational Methods and Clinical Applications for Spine Imaging*, pages 235–240. Springer, 2015.

[139] S.B. Kotsiantis, I. Zaharakis, and P. Pintelas. Supervised machine learning: A review of classification techniques, 2007.

[140] S.B. Kotsiantis, I. Zaharakis, and P. Pintelas. Supervised machine learning: A review of classification techniques, 2007.

[141] A. Krull, E. Brachmann, F. Michel, M. Yin, S. Gumhold, and C. Rother. Learning analysis-by-synthesis for 6d pose estimation in rgb-d images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 954–962, 2015.

[142] U. Kurkure, O.C. Avila-Montes, and I.A. Kakadiaris. Automated segmentation of thoracic aorta in non-contrast ct images. In *Biomedical Imaging: From Nano to Macro, 2008. ISBI 2008. 5th IEEE International Symposium on*, pages 29–32. IEEE, 2008.

[143] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521:436–444, 2015.

[144] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[145] L.K. Lee, S.C. Liew, and W.J. Thong. A review of image segmentation methodologies in medical image. In *Advanced computer and communication engineering technology*, pages 1069–1080. Springer, 2015.

[146] S. Lee, G. Wolberg, and S.Y. Shin. Scattered data interpolation with multilevel b-splines. *Trans. Visualization and Computer Graphics*, 183(3), 1997.

[147] G. Lemaıtre and M. Radojevic. Directed reading: Boosting algorithms. *Heriot-Watt University, Universitat de Girona, Universite de Bourgogne*, 2009.

[148] M.B. Leon, C.R. Smith, M. Mack, D.C. Miller, J.W. Moses, L.G. Svensson, E.M. Tuzcu, J.G. Webb, G.P. Fontana, R.R. Makkar, et al. Transcatheter aortic-valve implantation for aortic stenosis in patients who cannot undergo surgery. *New England Journal of Medicine*, 363(17):1597–1607, 2010.

[149] V. Lepetit and P. Fua. Keypoint recognition using randomized trees. *IEEE transactions on pattern analysis and machine intelligence*, 28(9):1465–1479, 2006.

[150] N. Lermé, F. Malgouyres, and L. Létocart. Reducing graphs in graph cut segmentation. In *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pages 3045–3048. IEEE, 2010.

[151] 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.

[152] K. Levenberg. A method for the solution of certain non–linear problems in least squares. *Quarterly of Applied Mathematics*, 1944.

[153] G. Li, X. Chen, F. Shi, W. Zhu, J. Tian, and D. Xiang. Automatic liver segmentation based on shape constraints and deformable graph cut in ct images. *Image Processing, IEEE Transactions on*, 24(12):5315–5329, 2015.

[154] J. Li and Y. Zhang. Learning surf cascade for fast and accurate object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3468–3475, 2013.

[155] K. Li, X. Wu, D.Z. Chen, and M. Sonka. Optimal surface segmentation in volumetric images-a graph-theoretic approach. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(1):119–134, 2006.

[156] J. Liang, T. McInerney, and D. Terzopoulos. United snakes [image analysis software]. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 933–940. IEEE, 1999.

[157] J. Liang, T. McInerney, and D. Terzopoulos. United snakes. *Medical image analysis*, 10(2):215–233, 2006.

[158] S.V. Lichtenstein, A. Cheung, J. Ye, C.R. Thompson, R.G. Carere, S. Pasupati, and J.G. Webb. Transapical transcatheter aortic valve implantation in humans initial clinical experience. *Circulation*, 114(6):591–596, 2006.

[159] H. Ling, S.K. Zhou, Y. Zheng, B. Georgescu, M. Suehling, and D. Comaniciu. Hierarchical, learning-based automatic liver segmentation. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[160] K. Liu, H. Skibbe, T. Schmidt, T. Blein, K. Palme, and O. Ronneberger. 3d rotation-invariant description from tensor operation on spherical hog field. *NeuroImage*, 57(2):416–422, 2011.

[161] T. Liu, H. Xu, W. Jin, Z. Liu, Y. Zhao, and W. Tian. Medical image segmentation based on a hybrid region-based active contour model. *Computational and mathematical methods in medicine*, 2014, 2014.

[162] H. Lombaert, Y. Sun, L. Grady, and C. Xu. A multilevel banded graph cuts method for fast image segmentation. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 259–265. IEEE, 2005.

[163] D.G. Lowe. Local feature view clustering for 3d object recognition. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–682. IEEE, 2001.

[164] D. Lu, X. Yu, X. Jin, B. Li, Q. Chen, and J. Zhu. Neural network based edge detection for automated medical diagnosis. In *IEEE ICIA*, pages 343–348. IEEE, 2011.

[165] P. Lurz, L. Coats, S. Khambadkone, J. Nordmeyer, Y. Boudjemline, S. Schievano, V. Muthurangu, T.Y. Lee, G. Parenzan, G. Derrick, et al. Percutaneous pulmonary valve implantation impact of evolving technology and learning curve on clinical outcome. *Circulation*, 117(15):1964–1972, 2008.

[166] J. Ma, L. Lu, Y. Zhan, X. Zhou, M. Salganicoff, and A. Krishnan. Hierarchical segmentation and identification of thoracic vertebra using learning-based edge detection and coarse-to-fine deformable model. In *MICCAI 2010*, pages 19–27. Springer, 2010.

[167] D. MacDonald, D. Avis, and A.C. Evans. Multiple surface identification and matching in magnetic resonance images. In *Visualization in Biomedical Computing 1994*, pages 160–169. International Society for Optics and Photonics, 1994.

[168] J.B.A. Maintz and M.A. Viergever. A survey of medical image registration. *Medical image analysis*, 2(1):1–36, 1998.

[169] R. Malladi and J.A. Sethian. Level set and fast marching methods in image processing and computer vision. In *Image Processing, 1996. Proceedings., International Conference on*, volume 1, pages 489–492. IEEE, 1996.

[170] R. Malladi, J.A. Sethian, and B.C. Vemuri. Shape modeling with front propagation: A level set approach. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(2):158–175, 1995.

[171] D.W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.

[172] M.H. Maurer, B. Hamm, and M. Dewey. Survey regarding the clinical practice of cardiac ct in germany: indications, scanning technique and reporting. *RoFo: Fortschritte auf dem Gebiete der Rontgenstrahlen und der Nuklearmedizin*, 181(12):1135–1143, 2009.

[173] T. McInerney and D. Terzopoulos. Deformable models in medical image analysis: A survey. *MIA*, 1(2):91–108, 1996.

[174] T. McInerney and D. Terzopoulos. Deformable models. In *Handbook of medical imaging*, pages 127–145. Academic Press, Inc., 2000.

[175] S. Menet, P. Saint-Marc, and G. Medioni. B-snakes: Implementation and application to stereo. In *proceedings DARPA*, volume 720, page 726, 1990.

[176] P. Mesejo, O. Ibáñez, O. Cordón, and S. Cagnoni. A survey on image segmentation using metaheuristic-based deformable models: state of the art and critical analysis. *Applied Soft Computing*, 44:1–29, 2016.

[177] F. Michel, A. Krull, E. Brachmann, M.Y. Yang, S. Gumhold, and C. Rother. Pose estimation of kinematic chain instances via object coordinate regression. In *Proc. British Machine Vision Conf*, pages 181–1, 2015.

[178] J.V. Miller, D.E. Breen, W.E. Lorensen, R.M. O'Bara, and M.J. Wozny. Geometrically deformed models: a method for extracting closed geometric models form volume data. In *ACM SIGGRAPH Computer Graphics*, volume 25, pages 217–226. ACM, 1991.

[179] M.O. Mohr and R. Berger. Towards autonomy in active contour models. 1990.

[180] J. Montagnat and H. Delingette. Volumetric medical images segmentation using shape constrained deformable models. In *CVRMed-MRCAS'97*, pages 13–22. Springer, 1997.

[181] J. Montagnat, H. Delingette, and N. Ayache. A review of deformable surfaces: topology, geometry and deformation. *Image and vision computing*, 19(14):1023–1040, 2001.

[182] A.M. Mood. Introduction to the theory of statistics. 1950.

[183] L.A. Moulopoulos and V. Koutoulidis. *Bone Marrow MRI: A Pattern-Based Approach*. Springer, 2014.

[184] E Munoz, Y. Konishi, V. Murino, and A. Del Bue. Fast 6d pose estimation for texture-less objects from a single rgb image. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5623–5630. IEEE, 2016.

[185] Y. Murai, Y. Yamauchi, T. Yamashita, and H. Fujiyoshi. Weighted hough forest for object detection. In *Machine Vision Applications (MVA), 2015 14th IAPR International Conference on*, pages 122–125. IEEE, 2015.

[186] K.H. Nicolaides, G. Rizzo, K. Hecher, and R. Ximenes. Doppler in obstetrics. *London: The Fetal Medicine Foundation*, 2002.

[187] Diagnostoc Imaging of Salem. Low-dose ct. `http://www.salemimaging.com/ct.php`. Online; accessed 15-05-2016.

[188] J. Ophir, S.K. Alam, B.S. Garra, F. Kallel, E.E. Konofagou, T. Krouskop, C.R.B. Merritt, R. Righetti, R. Souchon, S. Srinivasan, et al. Elastography: imaging the elastic properties of soft tissues with ultrasound. *Journal of Medical Ultrasonics*, 29(4):155–171, 2002.

[189] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: an application to face detection. In *Computer vision and pattern recognition, 1997. Proceedings., 1997 IEEE computer society conference on*, pages 130–136. IEEE, 1997.

[190] KFSP Tumor Oxygenation. Positron emission tomography (pet). `http://www.to2.uzh.ch/researchprojects/projectbuckweber.html`. Online; accessed 15-05-2016.

[191] M. Ozuysal, M. Calonder, V. Lepetit, and P. Fua. Fast keypoint recognition using random ferns. *IEEE transactions on pattern analysis and machine intelligence*, 32(3):448–461, 2010.

[192] M. Ozuysal, P. Fua, and V. Lepetit. Fast keypoint recognition in ten lines of code. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. Ieee, 2007.

[193] N.R. Pal and S.K. Pal. A review on image segmentation techniques. *Pattern recognition*, 26(9):1277–1294, 1993.

[194] J.B. Pawley and B.R. Masters. Handbook of biological confocal microscopy. *Optical Engineering*, 35(9):2765–2766, 1996.

[195] N. Perveen, D. Kumar, and I. Bhardwaj. An overview on template matching methodologies and its applications. *IJRCCT*, 2(10):988–995, 2013.

[196] J. Peters, O. Ecabert, C. Lorenz, J. von Berg, M.J. Walker, T.B. Ivanc, M. Vembar, M.E. Olszewski, and J. Weese. Segmentation of the heart and major vascular structures in cardiovascular ct images. In *Medical Imaging*, pages 691417–691417. International Society for Optics and Photonics, 2008.

[197] J. Peters, O. Ecabert, C. Meyer, H. Schramm, R. Kneser, A. Groth, and J. Weese. Automatic whole heart segmentation in static magnetic resonance image volumes. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2007*, pages 402–410. Springer, 2007.

[198] C. Petitjean and J.N. Dacher. A review of segmentation methods in short axis cardiac mr images. *Medical image analysis*, 15(2):169–184, 2011.

[199] M. Petrou and C. Petrou. *Image processing: the fundamentals*. John Wiley & Sons, 2010.

[200] M.E. Phelps. *PET: molecular imaging and its biological applications.* Springer Science & Business Media, 2004.

[201] A. Pratondo, C.K. Chui, and S.H. Ong. Robust edge-stop functions for edge-based active contour models in medical image segmentation. *Signal Processing Letters, IEEE*, 23(2):222–226, 2016.

[202] X. Qian, J. Wang, S. Guo, and Q. Li. An active contour model for medical image segmentation with application to brain ct image. *Medical physics*, 40(2):021911, 2013.

[203] H. Quigley, S.J. Colloby, and J.T. O'Brien. Pet imaging of brain amyloid in dementia: a review. *International journal of geriatric psychiatry*, 26(10):991–999, 2011.

[204] M. Rajadhyaksha, M. Grossman, D. Esterowitz, R.H. Webb, and R.R. Anderson. In vivo confocal scanning laser microscopy of human skin: melanin provides strong contrast. *Journal of Investigative Dermatology*, 104(6):946–952, 1995.

[205] J.C. Rasmussen, I.C. Tan, M.V. Marshall, C.E. Fife, and E.M. Sevick-Muraca. Lymphatic imaging in humans with near-infrared fluorescence. *Current opinion in biotechnology*, 20(1):74–82, 2009.

[206] A. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 806–813, 2014.

[207] T.R. Reed and J.M.H. Dubuf. A review of recent texture segmentation and feature extraction techniques. *CVGIP: Image understanding*, 57(3):359–372, 1993.

[208] D. Resnick and M.J. Kransdorf. *Bone and joint imaging.* Elsevier Saunders Philadelphia, 2005.

[209] R. Rios-Cabrera and T. Tuytelaars. Discriminatively trained templates for 3d object detection: A real time scalable approach. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2048–2055, 2013.

[210] J. Rissanen. *Minimum description length principle.* Wiley Online Library, 1985.

[211] J. Rissanen. *Stochastic complexity in statistical inquiry*, volume 15. World scientific, 1998.

[212] J. Rogowska. Overview and fundamentals of medical image segmentation. In *Handbook of medical imaging*, pages 69–85. Academic Press, Inc., 2000.

[213] W.C. Röntgen. On a new kind of rays. *Science*, pages 227–231, 1896.

[214] E. Rosten, R. Porter, and T. Drummond. Faster and better: A machine learning approach to corner detection. *IEEE transactions on pattern analysis and machine intelligence*, 32(1):105–119, 2010.

[215] D. Rueckert, A.F. Frangi, and J.A. Schnabel. Automatic construction of 3-d statistical deformation models of the brain using nonrigid registration. *Medical Imaging, IEEE Transactions on*, 22(8):1014–1025, 2003.

[216] D. Rueckert, L.I. Sonoda, C. Hayes, D.L.G. Hill, M.O. Leach, and D.J. Hawkes. Nonrigid registration using free-form deformations: application to breast mr images. *Medical Imaging, IEEE Transactions on*, 18(8):712–721, 1999.

[217] Z. Rumboldt, M. Castillo, B. Huang, and A. Rossi. *Brain imaging with MRI and CT: An image pattern approach.* Cambridge University Press, 2012.

[218] D.E. Rumelhart, G.E. Hinton, and R.J. Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1, 1988.

[219] Y. Sahillioğlu and Y. Yemez. 3d shape correspondence by isometry-driven greedy optimization.

[220] B. Sahiner, H.P. Chan, N. Petrick, D. Wei, M.A. Helvie, D.D. Adler, and M. M. Goodsitt. Classification of mass and normal breast tissue: a convolution neural network classifier with spatial domain and texture images. *Medical Imaging, IEEE Transactions on*, 15(5):598–610, 1996.

[221] R.E. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. *Machine learning*, 37(3):297–336, 1999.

[222] G.P. Schmidt, S.O. Schoenberg, M.F. Reiser, and A. Baur-Melnyk. Whole-body mr imaging of bone marrow. *European journal of radiology*, 55(1):33–40, 2005.

[223] H. Schneiderman and T. Kanade. Object detection using the statistics of parts. *International Journal of Computer Vision*, 56(3):151–177, 2004.

[224] M. Schwarz, H. Schulz, and S. Behnke. Rgb-d object recognition and pose estimation based on pre-trained convolutional neural network features. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1329–1335. IEEE, 2015.

[225] T.W. Sederberg and S.R. Parry. Free-form deformation of solid geometric models. *ACM SIGGRAPH computer graphics*, 20(4):151–160, 1986.

[226] M.M. Selim, K.M. Kelly, J.S. Nelson, G. Wendelschafer-Crabb, W.R. Kennedy, and B.D. Zelickson. Confocal microscopy study of nerves and blood vessels in untreated and treated port wine stains: preliminary observations. *Dermatologic surgery*, 30(6):892–897, 2004.

[227] N. Senthilkumaran and R. Rajesh. Edge detection techniques for image segmentation–a survey of soft computing approaches. *IJRTE*, 1(2), 2009.

[228] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun. Pedestrian detection with unsupervised multi-stage feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3626–3633, 2013.

[229] J.A. Sethian. *Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*, volume 3. Cambridge university press, 1999.

[230] L. Shafarenko, M. Petrou, and J. Kittler. Automatic watershed segmentation of randomly textured color images. *Image Processing, IEEE Transactions on*, 6(11):1530–1544, 1997.

[231] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 806–813, 2014.

[232] T. Shi and S. Horvath. Unsupervised learning with random forest predictors. *Journal of Computational and Graphical Statistics*, 2012.

[233] H.C. Shin, M.R. Orton, D.J. Collins, S.J. Doran, and M.O. Leach. Stacked autoencoders for unsupervised feature learning and multiple organ detection in a pilot study using 4d patient data. *IEEE T-PAMI*, 35(8):1930–1943, 2013.

[234] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[235] H. Skibbe, M. Reisert, and H. Burkhardt. Shog-spherical hog descriptors for rotation invariant 3d object detection. In *Pattern Recognition*, pages 142–151. Springer, 2011.

[236] E. Smistad, T.L. Falch, M. Bozorgi, A.C. Elster, and F. Lindseth. Medical image segmentation on gpus–a comprehensive review. *Medical image analysis*, 20(1):1–18, 2015.

[237] J. Snyman. *Practical mathematical optimization: an introduction to basic optimization theory and classical and new gradient-based algorithms*, volume 97. Springer Science & Business Media, 2005.

[238] M. Sonka, V. Hlavac, and R. Boyle. *Image processing, analysis, and machine vision.* Cengage Learning, 2014.

[239] A. Sotiras, C. Davatzikos, and N. Paragios. Deformable medical image registration: A survey. *Medical Imaging, IEEE Transactions on*, 32(7):1153–1190, 2013.

[240] A. Srinivasan and S. Sundaram. Applications of deformable models for in-dopth analysis and feature extraction from medical images—a review. *Pattern recognition and image analysis*, 23(2):296–318, 2013.

[241] L.H. Staib and J.S. Duncan. Boundary finding with parametrically deformable models. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (11):1061–1075, 1992.

[242] C. Steger. Similarity measures for occlusion, clutter, and illumination invariant object recognition. In *Joint Pattern Recognition Symposium*, pages 148–154. Springer, 2001.

[243] M.B. Stegmann. *Active appearance models.* PhD thesis, PhD thesis, Technical University of Denmark, 2000. 125, 129, 2007.

[244] M. Styner, I. Oguz, T. Heimann, and G. Gerig. Minimum description length with local geometry. In *Biomedical Imaging: From Nano to Macro, 2008. ISBI 2008. 5th IEEE International Symposium on*, pages 1283–1286. IEEE, 2008.

[245] G. Subsol, J.P. Thirion, and N. Ayache. A scheme for automatically building three-dimensional morphometric anatomical atlases: application to a skull atlas. *Medical Image Analysis*, 2(1):37–60, 1998.

[246] M. Sussman and E.G. Puckett. A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows. *Journal of Computational Physics*, 162(2):301–337, 2000.

[247] P.N. Tan, M. Steinbach, V. Kumar, et al. *Introduction to data mining*, volume 1. Pearson Addison Wesley Boston, 2006.

[248] G. Taubin. A signal processing approach to fair surface design. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 351–358. ACM, 1995.

[249] A. Tejani, D. Tang, R. Kouskouridas, and T.K. Kim. Latent-class hough forests for 3d object detection and pose estimation. In *European Conference on Computer Vision*, pages 462–477. Springer, 2014.

[250] D. Terzopoulos. On matching deformable models to images. In *Topical Meeting on Machine Vision Tech. Digest Series*, volume 12, pages 160–167, 1987.

[251] D. Terzopoulos and K. Fleischer. Deformable models. *The visual computer*, 4(6):306–331, 1988.

[252] D. Terzopoulos, A. Witkin, and M. Kass. Constraints on deformable models: Recovering 3d shape and nonrigid motion. *Artificial intelligence*, 36(1):91–123, 1988.

[253] D. Terzopoulos, A. Witkin, and M. Kass. Constraints on deformable models: Recovering 3d shape and nonrigid motion. *Artificial intelligence*, 36(1):91–123, 1988.

[254] H.H. Thodberg. Minimum description length shape and appearance models. In *Information Processing in Medical Imaging*, pages 51–62. Springer, 2003.

[255] P.A. Tipler and G. Mosca. *Physics for scientists and engineers.* Macmillan, 2007.

[256] G. Tsechpenakis. Deformable model-based medical image segmentation. In *Multi Modality State-of-the-Art Medical Image Segmentation and Registration Methodologies*, pages 33–67. Springer, 2011.

[257] Z. Tu. Probabilistic boosting-tree: Learning discriminative models for classification, recognition, and clustering. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1589–1596. IEEE, 2005.

[258] B.J. Vakoc, R.M. Lanning, J.A. Tyrrell, T.P. Padera, L.A. Bartlett, T. Stylianopoulos, L.L. Munn, G.J. Tearney, D. Fukumura, R.K. Jain, et al. Three-dimensional microscopy of the tumor microenvironment in vivo using optical frequency domain imaging. *Nature medicine*, 15(10):1219–1223, 2009.

[259] B. Van Ginneken, A. Frangi, J.J. Staal, B.M. Romeny, and M.A. Viergever. Active shape model segmentation with optimal features. *Medical Imaging, IEEE Transactions on*, 21(8):924–933, 2002.

[260] M.W. Vernooij, M.A. Ikram, H.L. Tanghe, A. Vincent, A. Hofman, G.P. Krestin, W.J. Niessen, M.M.B. Breteler, and A. van der Lugt. Incidental findings on brain mri in the general population. *New England Journal of Medicine*, 357(18):1821–1828, 2007.

[261] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511. IEEE, 2001.

[262] I. Waechter, R. Kneser, G. Korosoglou, J. Peters, N. Bakker, Rvd. Boomen, and J. Weese. Patient specific models for planning and guidance of minimally invasive aortic valve implantation. *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2010*, pages 526–533, 2010.

[263] R.J. Wakefield, P.V. Balint, M. Szkudlarek, E. Filippucci, M. Backhaus, M.A. D'Agostino, E.N. Sanchez, A. Iagnocco, W.A. Schmidt, G.A.W. Bruyn, et al. Musculoskeletal ultrasound including definitions for ultrasonographic pathology. *The Journal of rheumatology*, 32(12):2485–2487, 2005.

[264] S. Wang, L. Fu, Y. Yue, Y. Kang, and J. Liu. Fast and automatic segmentation of ascending aorta in msct volume data. In *Image and Signal Processing, 2009. CISP'09. 2nd International Congress on*, pages 1–5. IEEE, 2009.

[265] Y. Wang, B.S. Peterson, and L.H. Staib. Shape-based 3d surface correspondence using geodesics and local geometry. In *Computer Vision and Pattern*

Recognition, 2000. Proceedings. IEEE Conference on, volume 2, pages 644–651. IEEE, 2000.

[266] H. Wechsler, J.P. Phillips, V. Bruce, F.F. Soulie, and T.S. Huang. *Face recognition: From theory to applications*, volume 163. Springer Science & Business Media, 2012.

[267] J. Weese, M. Kaus, C. Lorenz, S. Lobregt, R. Truyen, and V. Pekar. Shape constrained deformable models for 3d medical image segmentation. In *Information Processing in Medical Imaging*, pages 380–387. Springer, 2001.

[268] O. Wieben, C. Francois, and S.B. Reeder. Cardiac mri of ischemic heart disease at 3t: potential and challenges. *European journal of radiology*, 65(1):15–28, 2008.

[269] M.C. Williams, J.H. Reid, G. McKillop, N.W. Weir, E.J.R. van Beek, N.G. Uren, and D.E. Newby. Cardiac and coronary ct comprehensive imaging approach in the assessment of coronary heart disease. *Heart*, 97(15):1198–1205, 2011.

[270] S. Winder, G. Hua, and M. Brown. Picking the best daisy. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 178–185. IEEE, 2009.

[271] A. Witkin, D. Terzopoulos, and M. Kass. Signal matching through scale space. *International Journal of Computer Vision*, 1(2):133–144, 1987.

[272] P Wohlhart and V Lepetit. Learning descriptors for object recognition and 3d pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3109–3118, 2015.

[273] BioOptics World. Confocal microscopy method effective as diagnostic tool for sarcoidosis. `http://www.biiopticsworld.com/articles.html`. Online; accessed 15-05-2016.

[274] X. Xie. Active contouring based on gradient vector interaction and constrained level set diffusion. *Image Processing, IEEE Transactions on*, 19(1):154–164, 2010.

[275] C. Xu, D.L. Pham, and J.L. Prince. Image segmentation using deformable models. *Handbook of medical imaging*, 2:129–174, 2000.

[276] C. Xu and J.L. Prince. Generalized gradient vector flow external forces for active contours. *Signal processing*, 71(2):131–139, 1998.

[277] C. Xu and J.L. Prince. Snakes, shapes, and gradient vector flow. *Image Processing, IEEE Transactions on*, 7(3):359–369, 1998.

[278] R. Yang, M. Mirmehdi, X. Xie, and D. Hall. Shape and appearance priors for level set-based left ventricle segmentation. *Computer Vision, IET*, 7(3):170–183, 2013.

[279] Yi. Yang and D. Ramanan. Articulated human detection with flexible mixtures of parts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(12):2878–2890, 2013.

[280] P.J. Yim, R. Cebral, R. Mullick, and P.L. Choyle. Vessel surface reconstruction with a tubular deformable model. *IEEE T-MI*, 20:1411–1421, 2001.

[281] H. Zhang, A. Sheffer, D. Cohen-Or, Q. Zhou, O. Van Kaick, and A. Tagliasacchi. Deformation-driven shape correspondence. In *Computer Graphics Forum*, volume 27, pages 1431–1439. Wiley Online Library, 2008.

[282] W. Zhang, J. Liu, J. Yao, and R.M. Summers. Segmenting the thoracic, abdominal and pelvic musculature on ct scans combining atlas-based model and active contour model. In *SPIE Medical Imaging*, pages 867008–867008. International Society for Optics and Photonics, 2013.

[283] Z. Zhang and M. Braun. Fully 3d active surface models with self-inflation and self-deflation forces. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 85–90. IEEE, 1997.

[284] L. Zheng, G. Li, and J. Sha. The survey of medical image 3d reconstruction. In *Fifth International Conference on Photonics and Imaging in Biology*

and Medicine, pages 65342K–65342K. International Society for Optics and Photonics, 2007.

[285] Y. Zheng, A. Barbu, B. Georgescu, M. Scheuering, and D. Comaniciu. Four-chamber heart modeling and automatic segmentation of for 3d cardiac ct volumes using marginal space learning and steerable features. *IEEE T-MI*, 27(11):1668–1681, 2008.

[286] Y. Zheng and D. Comaniciu. Marginal space learning. In *Marginal Space Learning for Medical Image Analysis*, pages 25–65. Springer, 2014.

[287] Y. Zheng, B. Georgescu, H. Ling, S Zhou, M. Scheuering, and D. Comaniciu. Constrained marginal space learning for efficient 3d anatomical structure detection in medical images. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 194–201. IEEE, 2009.

[288] Y. Zheng, M. John, R. Liao, A. Nottling, J. Boese, J. Kempfert, T. Walther, G. Brockmann, and D. Comaniciu. Automatic aorta segmentation and valve landmark detection in c-arm ct for transcatheter aortic valve implantation. *IEEE T-MI*, 31:2307–2321, 2012.

[289] Y. Zheng, X. Lu, B. Georgescu, A. Littmann, E. Mueller, and D. Comaniciu. Automatic left ventricle detection in mri images using marginal space learning and component-based voting. In *SPIE Medical Imaging*, pages 725906–725906. International Society for Optics and Photonics, 2009.

[290] Y. Zheng, X. Lu, B. Georgescu, A. Littmann, E. Mueller, and D. Comaniciu. Robust object detection using marginal space learning and ranking-based multi-detector aggregation: Application to left ventricle detection in 2d mri images. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1343–1350. IEEE, 2009.

[291] Y.P. Zheng and A.F.T. Mak. An ultrasound indentation system for biomechanical properties assessment of soft tissues in-vivo. *Biomedical Engineering, IEEE Transactions on*, 43(9):912–918, 1996.

[292] S. Zhou, J. Wang, S. Zhang, Y. Liang, and Y. Gong. Active contour model based on local and global intensity information for medical image segmentation. *Neurocomputing*, 2016.

[293] W. Zhu, S. Ha Kang, and G. Biros. A geodesic-active-contour-based variational model for short-axis cardiac mr image segmentation. *International Journal of Computer Mathematics*, 90(1):124–139, 2013.