SEGMENTATION AND RECONSTRUCTION OF 3D ARTERY MODELS FOR SURGICAL PLANNING

QI YINGYI

(B.Sc., FUDAN UNIVERSITY, 2005)

A THESIS SUBMITTED

FOR THE DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

SCHOOL OF COMPUTING

NATIONAL UNIVERSITY OF SINGAPORE

2008

Acknowledgements

I would like to express my deep and sincere gratitude to my supervisor, Assoc. Prof. Leow Wee Kheng. His wide knowledge and his logical way of thinking have been a great value for me. His understanding and guidance have provided a good basis of my thesis work. His enthusiasm in research has encouraged me a lot. I also thank my colleagues: Ding Feng, Li hao, Song Zhiyuan, Zhang Xiaopeng, etc. I really appreciate the help they gave me in the past three years. I really enjoyed the pleasant stay with these brilliant people at Computer Vision Lab of School of Computing (SOC), NUS. Finally, I would like to thank my family for their endless love and support.

QI YingYi

NATIONAL UNIVERSITY OF SINGAPORE

March, 2008

Contents

Abstract

This thesis focuses on segmentation and 3D reconstruction of cardiac arteries from CT images of patients with TGA. The purpose of this work is to build 3D models for surgical planning and simulation system. This thesis proposes different algorithms to segment the great arteries and the coronary arteries according to different characteristics of arteries. A two-phase level set algorithm segments the great arteries. A geometric parametric model based algorithm segments the coronary arteries. Then, marching cubes algorithm is applied to reconstruct 3D models based on the segmentation results. The system was tested on three data sets — two TGA cases and one normal case. The results of segmentation and reconstruction are analyzed and discussed.

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Transposition of the great arteries (TGA) is a malformation in which the two great arteries, i.e., aorta and pulmonary trunk, connect to the wrong heart chambers (Figure 1.1). The pulmonary trunk, which normally carries de-oxygenated blood from the right ventricle to the lungs, now arises from the left ventricle. The aorta, which normally carries oxygenated blood from the left ventricle to the rest of the body, arises instead from the right ventricle. TGA is a heart defect present at birth. Infants born with TGA cannot survive for a long time because they will not have enough oxygen in the bloodstream to meet the body's demands.

Transposition of the great arteries can be surgically repaired by a procedure called arterial switch operation (ASO). In the surgery, aorta, pulmonary trunk, and coronary arteries are moved to their normal positions to restore normal circulation of the blood through the heart and the lungs. The surgery is a very complex and difficult operation. It requires delicate surgical techniques because the arteries of newborn babies are usually very thin. It requires detailed planning for each case respectively because the anatomical structures of the cardiac vessels vary a lot

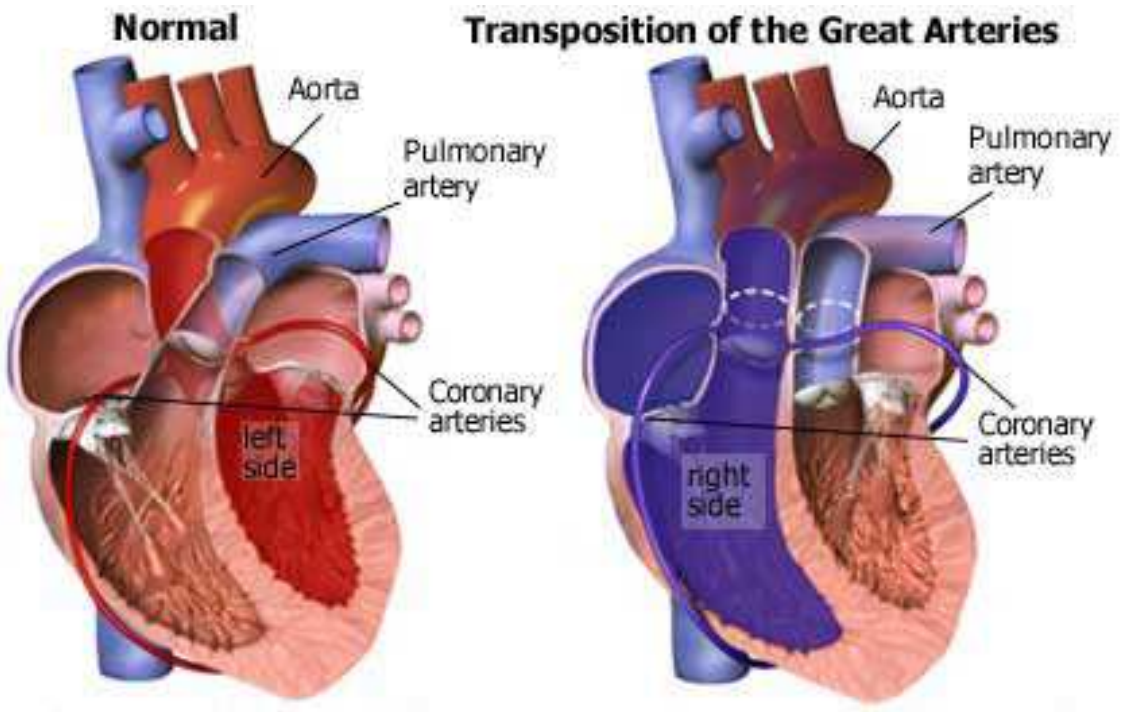


Figure 1.1: The anatomy of the normal heart and the heart with TGA (taken from http://heart.health.ivillage.com/signssympotoms/bluebaby3.cfm).

among different patients.

At present, there is no computerized tools for ASO planning. The surgeons develop the surgical plan based on CT images of the patient's heart. The plans are drawn by hand to show the surgical details. This planning method is not precise.

A computer-aided system can help the surgeons to perform surgery planning. It can construct 3D cardiac model from patient's CT images, and visualize it in the computer monitor. It can simulate the surgical plans under the direction of the surgeons. It can also show the expected surgical results and allow the surgeons to evaluate the results to determine the best plan.

1.2 Thesis Objective

A computer system for the simulation and planning of ASO has to contain two parts. The first part performs segmentation and 3D reconstruction from cardiac CT images. The second part performs virtual surgical operations based on the results provided by the first part.

The objective of this thesis project is to develop a system for segmenting and reconstructing 3D models of the great arteries and coronary arteries from CT images. The requirements of the system include the following:

1. The system should provide a correct segmentation result of the aorta, pulmonary trunk and coronary arteries.

2. The system should provide a complete 3D mesh model of arteries. The relative positions, orientation and sizes of arteries should be accurate enough for the purpose of surgical simulation.

The contributions of this thesis are as follows. First, it implemented a set of algorithms for segmenting and reconstructing 3D models of the great arteries and

coronary arteries. Second, it discussed the limitations of the segmentation algorithms and how to overcome them.

1.3 Thesis Organization

In order to develop algorithms to segment cardiac arteries, fundamental knowledge about the heart anatomy is required. Anatomy of a healthy heart and a heart with TGA will be discussed in Chapter 2, which includes a discussion of the structure of the arteries and their appearance in cardiac CT images. Existing segmentation algorithms, in particular vascular structure segmentation algorithm will be discussed in Chapter 3. According to different characteristics of the arteries, different algorithms are proposed to segment the great arteries and the coronary arteries (Chapter 4). Segmentation and reconstruction results of different patients (with and without TGA) will be shown and analyzed in Chapter 4. Finally, conclusions of the whole thesis are made in Chapter 5.

Chapter 2

Anatomy of the Heart

2.1 Normal Heart

The heart is the organ that supplies blood and oxygen to all parts of the body. Heart has four chambers. The upper chambers are called the left and right atria, and the lower chambers are called the left and right ventricles (Figure 2.1). Blood is pumped away from the heart through arteries and returns to the heart through veins. There are three types of cardiac arteries, e.g., aorta, pulmonary trunk and coronary arteries (Figure 2.1).

Aorta is the largest artery in the human body, originating from the left ventricle. Aorta includes three parts. Ascending aorta is the section between the heart and aortic arch. Aortic arch is the peak part that looks like an inverted "U". Descending aorta runs from the aortic arch down to the body. It is located behind the heart chambers. So, it cannot be seen in Figure 2.1. Aortic valve is located at the junction between the left ventricle and the aorta (Figure 2.2).

Pulmonary trunk arises from the right ventricle and ascends in front of the aorta. It branches into the left and right pulmonary arteries. pulmonary trunk is short and wide. Pulmonary valve is located at the junction between the right ventricle

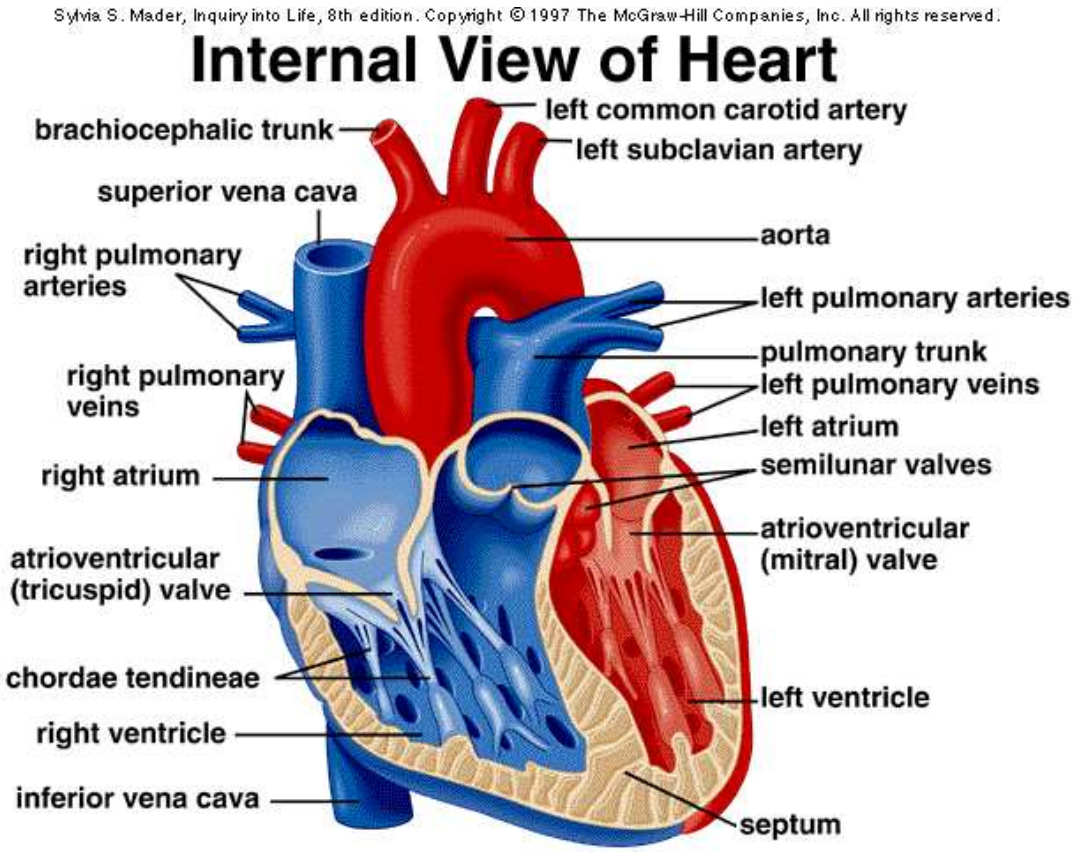


Figure 2.1: The anatomy of normal heart (taken from [Mad97]).

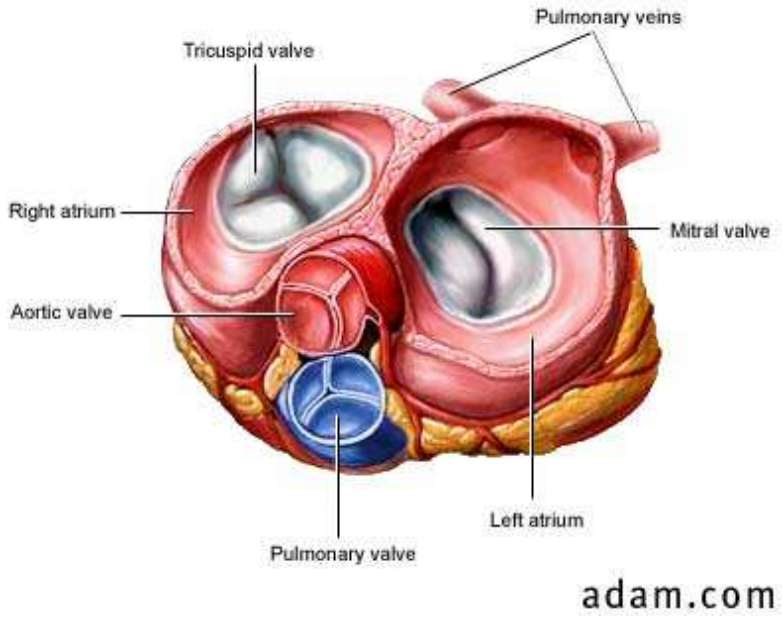


Figure 2.2: Superior view of the valves (taken from http://www.besthealth.com/besthealth/bodyguide/reftext/html/cardio_sys_fin.html).

and the pulmonary trunk.

Coronary arteries originate from the beginning of the ascending aorta, immediately above the aortic valve. The exact anatomy of the coronary artery varies considerably from person to person.

In CT images, arteries and chambers are always highlighted in white color by the contrast agent. The aortic arch appears as an ellipse in CT images (Figure 2.3(a)). Then it splits into the ascending aorta and descending aorta. Both of them are more like circles (Figure 2.3(b)). The ascending aorta is located in the middle of the heart. The descending aorta is located very close to the spine.

The top part of the pulmonary trunk is a T-shaped trunk. It appears as a curved tube in the CT images (Figure 2.3(c)). The trunk is very short and wide. Therefore, after only a few slices in the CT volume, the pulmonary trunk becomes a circle (Figure 2.3(d)). At this time, the pulmonary artery should be located in

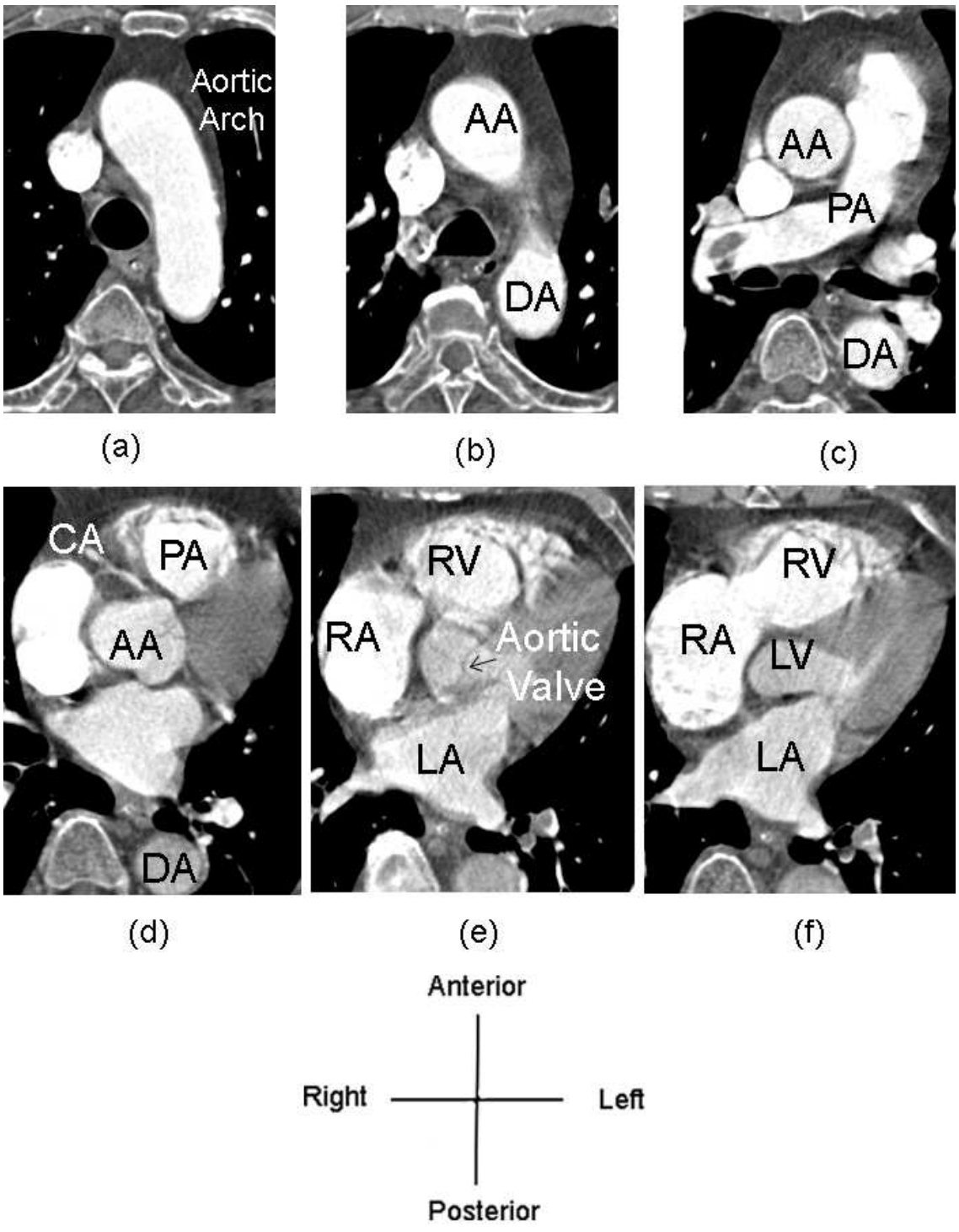


Figure 2.3: CT images of normal heart. (a)-(f): From top to bottom of the heart. AA: ascending aorta, DA: descending aorta, PA: pulmonary trunk, CA: coronary arteries, LA: left atrium, RA: right atrium, LV: left ventricle, and RV: right ventricle.

front of the ascending aorta in normal case.

Figure 2.3(e) and (f) show the relative positions of the four chambers. According to the positions of the chambers and the arteries, it can be seen that the pulmonary artery connects with the right ventricle, and the ascending aorta connects with the left ventricle. The valves between the arteries and the ventricles are located at their boundaries. They contain three radial lines that correspond to the valve leaflets (Figure 2.3(e)).

Coronary arteries are located near the root of the ascending aorta. They look like elongated tubes in CT images (Figure 2.3(d)). They are very thin. The diameters are usually smaller than the gaps between two slices. Therefore, the coronary arteries do not always show up in CT images.

2.2 Transposition of the Great Arteries

In the case of transportation of great arteries (TGA), the aorta and pulmonary trunk are connected to the wrong ventricles. Figure 2.4 shows the CT images of a TGA case. The aortic arch still appears as an ellipse (Figure 2.4(a)). Then, it branches into the ascending aorta and the descending aorta (Figure 2.4(b)). The pulmonary trunk is located near the ascending aorta (Figure 2.4(c,d)). Coronary arteries are located at the root of the ascending aorta as in the normal case (Figure 2.4(d)). According to the positions of the arteries and the ventricles (Figure 2.4(d,e)), it can be observed that the ascending aorta connects to the right ventricle and the pulmonary aorta connects to the left ventricle. This is different from the normal case.

Another distinguishing characteristic of TGA is the difference in the relative positions of the aorta and the pulmonary trunk. Normally, the pulmonary trunk lies in front of the ascending aorta (Figure 2.3(d)). But in TGA case, the pulmonary

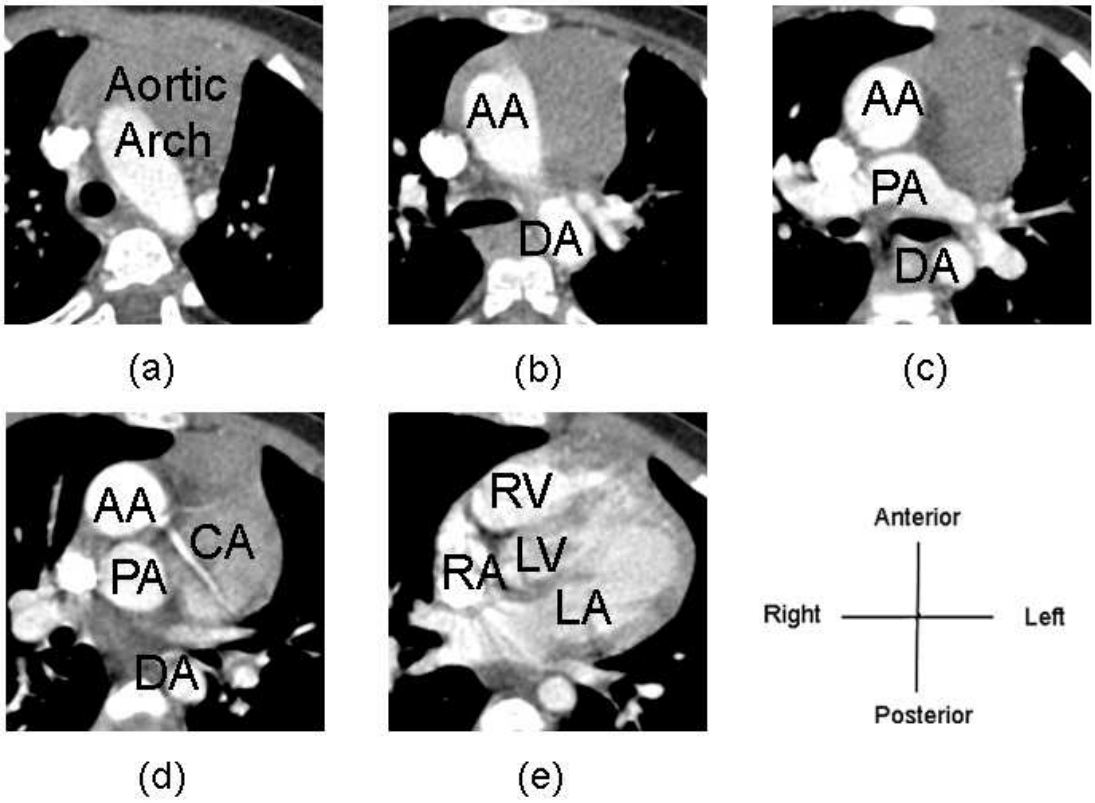


Figure 2.4: CT images of a patient with TGA. (a)-(e): From top to bottom of the heart. AA: ascending aorta, DA: descending aorta, PA: pulmonary trunk, CA: coronary arteries , LA: left atrium, RA: right atrium, LV: left ventricle, and RV: right ventricle.

trunk is located behind the aorta (Figure 2.4(d)).

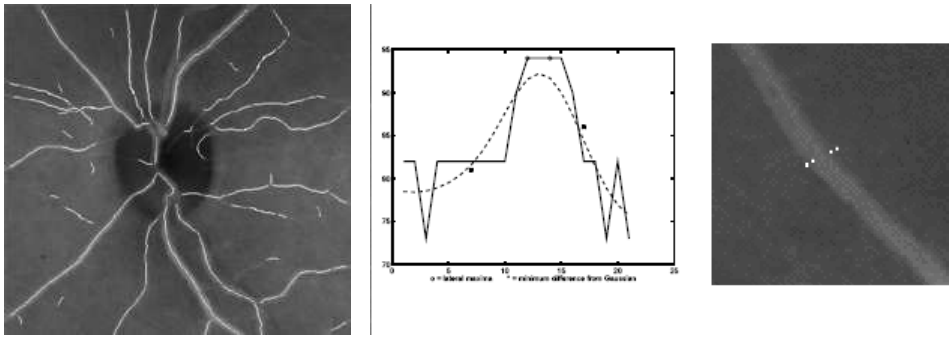
Chapter 3

Literature Review

There are several types of general segmentation algorithms, such as thresholding [Cas96, Dav97, KIF85], watershed [GW02, HEMK98, SSV+97], region growing [ESS+04, PT01, RG01], classification [SLC+05, dBvGB+03, KRFC05, HSD73] and clustering [IGR+04, WL90, GL00]. They can work well with simple biomedical images with homogeneous regions. On the other hand, complex medical images such as CT images usually have many inhomogeneous regions. Take CT image of vascular structure as an example. Pixels inside the vasculature do not necessarily have the same intensity. There is always some noise that appears as pixels with low intensities. In these cases, general segmentation algorithms may fail. Therefore, more sophisticated algorithms have been developed. This chapter reviews algorithms developed for segmenting vasculature and the heart.

3.1 Vasculature Segmentation Algorithms

Vasculatures are objects in the human body. They have tree structures with branches. Each branch often has a tubular shape. Vascular structures include blood vessels, coronary artery, neurovascular structures, airway tree (pulmonary



(a) (b) (c)

Figure 3.1: Tracked centerlines of retinal blood vessels (taken from [CPFT98]).

tree) and nerve channels. Due to the geometric and topological features of vascular structure, specific segmentation algorithms have been designed, such as centerline detection, snake, level set and geometric parametric model.

3.1.1 Centerline Detection

The main idea of the centerline detection approach is to find the centerlines of the entire vascular structure. One centerline detection method is the ridge-based algorithm [CPFT98, FMGW04]. It regards a gray-scale image as a 3D elevation map where intensity ridges approximate the skeleton of the vasculature. Ridge points, which are local peaks in intensity, are first detected. Then, the whole vascular structure is extracted by connecting neighboring ridge points (Figure 3.1).

Through centerline extraction, topological information of the vascular structure is obtained. It does not require special initialization. But, due to sensitivity to noise, it is difficult for centerline detection methods to extract all the small vessels.

3.1.2 Snake

Snake, also called active contour model, was first proposed by Kass, Witkin, and Terzopoulos [KWT87]. Snake is a contour represented by connected points. It behaves like a rubber band and can be deformed to match any shape under the influence of internal force, image force and external force. These three kinds of forces are defined so that the snake will conform to an object boundary or other features in the image. The internal energy is associated with elasticity and rigidity of the snake. The image energy specifies the image feature, such as edges, used to attract the snake. The external energy is not often used, and it can be spring energy, repulsion and so on. When the sum of these three kinds of energy is minimized, the snake should approach the desired image features.

Snake is a good model which can be used in many applications. Snake will fit the contour of any shape as long as the forces are well designed and balanced. It can connect disjoint edges by finding an optimal compromise between different forces. Moreover, it guarantees a smooth and closed boundary of the desired object. However, it has some intrinsic shortcomings. For instance, it can be very sensitive to initialization and noise, and it cannot fit well to concave features (Figure 3.2) [XP00].

To overcome these shortcomings, Xu and Prince proposed the gradient vector flow (GVF) method [XP00]. GVF field is a vector field derived from the diffusion of the gradient vectors of a gray-level or binary edge map computed from the input image. GVF replaces the usual image forces, and it can attract the snake to fit the concave part of the object in the image (Figure 3.2). Although GVF is less sensitive to initialization than traditional snake, it still requires a good initialization. It can also be distracted by noise.

Another variation is the dual snakes [GN94]. In this case, two snakes are used.

One is an interior snake lying within the regions of the desired object. The other is an exterior snake outside the desired regions. The two snakes are coupled using spring energy that causes them to be attracted to each other as well as the boundary of the desired object. Dual snake model incorporates the information from two snakes approaching the desired boundary from both sides. It reduces sensitivity to initialization. It also provides good performance in the case of non-convex shapes.

Snake can also be extended to three dimensions, known as active surface. In 3D, a surface is usually represented by a set of corrected control points, i.e. a surface mesh. Internal force, image force and external force are defined in a similar way as 2D snake. The computation of active surface algorithm is more complex than 2D snake. The number of control points are usually dramatically larger than that in 2D case. The structure of a surface mesh is more complex than a sequence of points in 2D case.

3.1.3 Level Set

Level set method was proposed by Osher and Sethian [OS88]. It represents the interface (2D contour or 3D surface) in one higher dimension. Consider, for example, the 2D contours Γ in Figure 3.3(a). A level set function $\phi(x, y, t)$ (the dark surface in Figure 3.3(b)) is defined in 3D. Typically, ϕ is the signed distance function to the contour. It keeps all possible states of Γ. The intersection of $\phi(x, y, t)$ and the x-y plane gives the contour Γ. Therefore, at any time t, Γ can be obtained by solving the equation $\phi(x, y, t) = 0$. The level set function $\phi(x, y, t)$ moves up and down the ϕ-axis under some predefined forces that give the propagation velocity of Γ.

Level set is used in many applications [MS96, QK01, Set96]. A simple 2D example is shown in Figure 3.4 [Set96]. Starting from a small circle within the vessel, the contour expands to fit the boundaries of the blood vessels. The idea of level

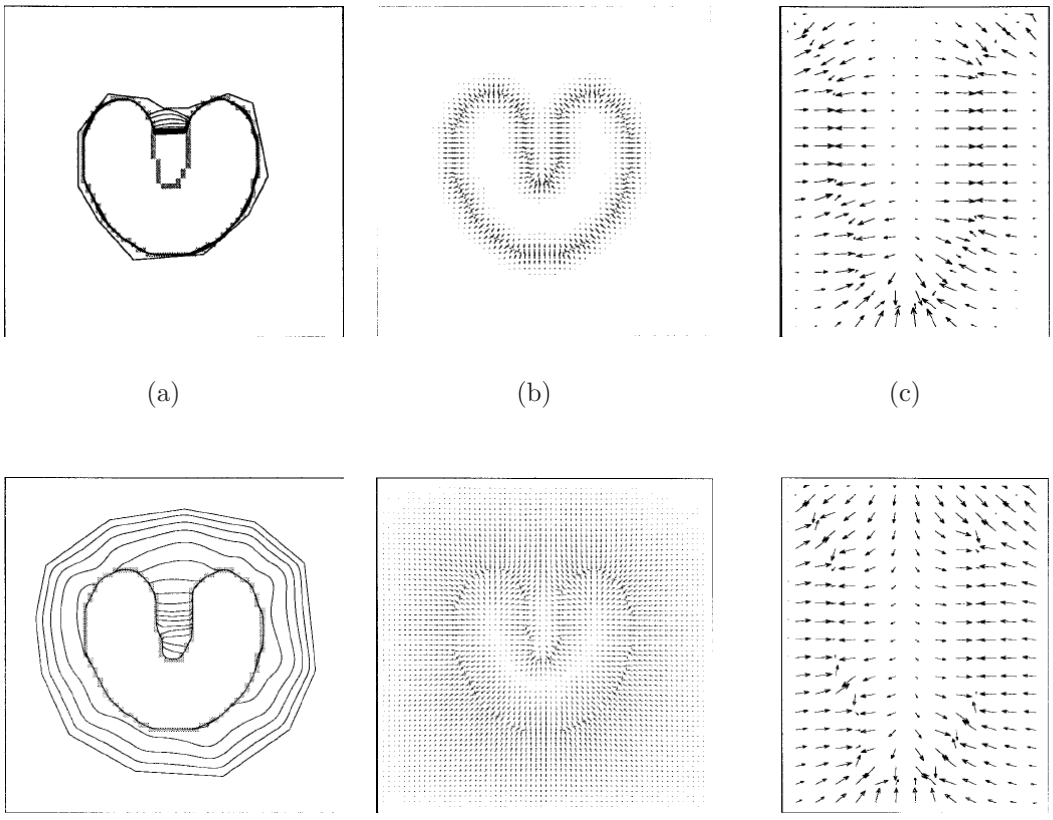
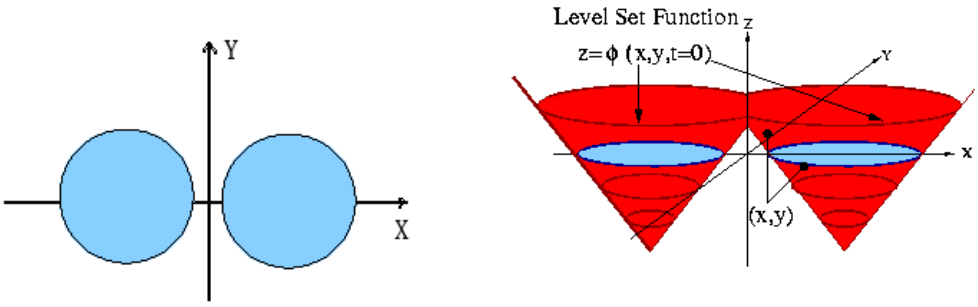


Figure 3.2: Comparison between traditional snake and GVF. (a) Convergence of a traditional snake, (b) traditional potential forces, (c) close-up of the concave part, (d) Convergence of a GVF snake, (e) GVF external forces, and (f) close-up of the concave part of GVF (taken from [XP98]).



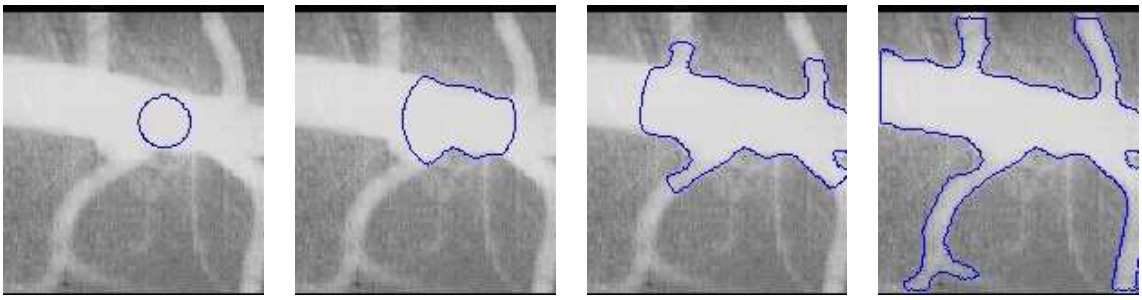
(a) (b)

Figure 3.3: Level set function. (a) 2D contours. (b) The level set surface (dark). The zero level set (light) gives the 2D contours (taken from [Leo]).

set can be easily extended to 3D case to extract surfaces of the desired objects [KQ03, LWK+05, NYT04].

The primary advantage of level set method is that the level set function remains a single function even when the zero level set changes topology, breaks, merges, or forms sharp corners when it evolves over time. Therefore, level set is useful for segmenting objects with complex topology such as blood vessels. However, traditional level set method does not contain geometrical constraints. So, the zero level set may leak into undesired regions.

To overcome the leakage problem, a level set model with a soft shape prior, called shape driven flow, is applied to segment 2D/3D vessels [NYT04] (Figure 3.5). Moreover, by including both intensity and shape information, this algorithm can overcome leakages near areas where image information is ambiguous.



(a) (b) (c) (d)

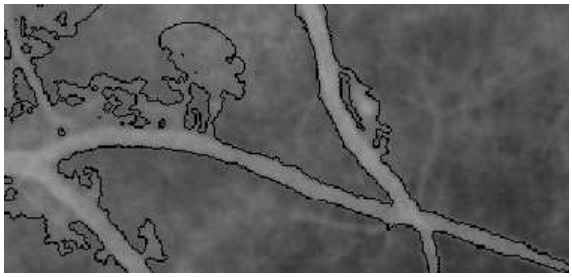
Figure 3.4: The evolution of the zero level set curve (taken from [Set96]). (a) The initial contour, (b, c) intermediate contours during evolution, and (d) the final contour of the blood vessel.

3.1.4 Geometric Parametric Model-based Approach

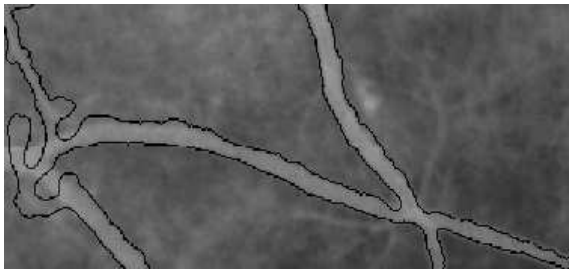
Vasculatures are tube-like objects. Consequently, the cross-section of each vessel can be approximated by an ellipse. Geometric parametric mode-based approach aligns a parametric elliptic model with vessel cross-section in every 2D slice of a 3D volumetric image to obtain a best fit [BP98, FIC04, PHS+94]. Each branch of the blood vessel is represented by a medial axis curve together with an elliptical surface that represents the vessel surface (Figure 3.6(a)). The location of the axis and the parameters of the ellipse are determined to match the model with the input image.

In [FIC04], two-stage deformation is applied to match the image. The first stage called axis deformation determines the position of each elliptical model for each vessel according to the medial axis (Figure 3.6(c)). The second stage performs surface deformation to determine the parameters of the elliptical model, such as semidiameter and orientation (Figure 3.6(d, e)).

Geometric parametric model-based approach is applicable to the segmentation of healthy vasculature. It does not have leakage problem because it always preserve the



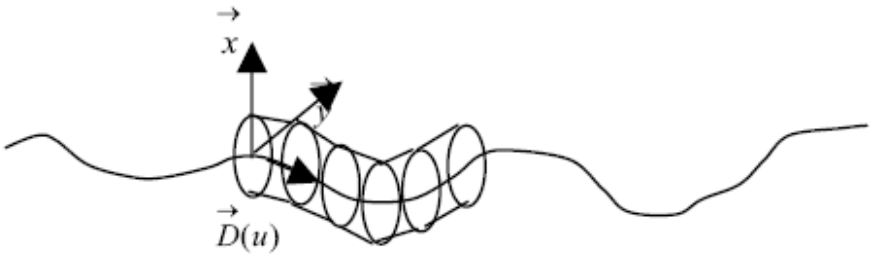
(a)



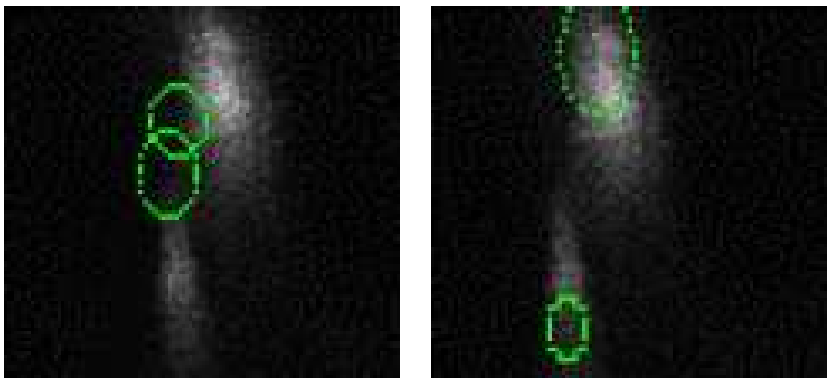
(b)

(c) (d) (e)

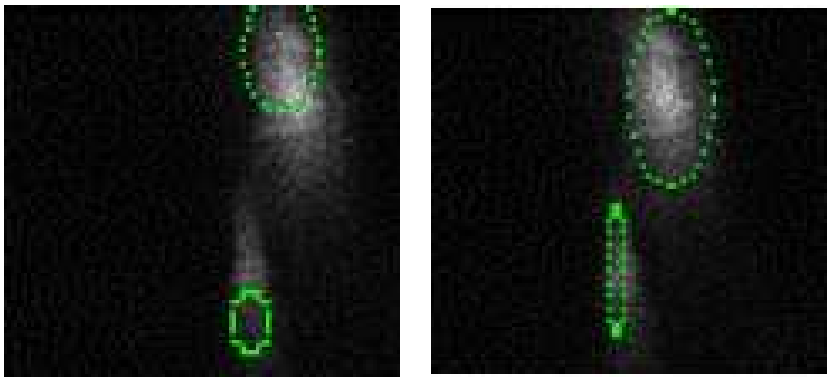
Figure 3.5: Level set segmentation using shape driven flow (taken from [NYT04]). (a) 2D result without shape prior. Contour leaks into several non-vessel regions. (b) 2D result with shape prior has a less severe leakage problem. (c) 3D results without shape prior. The leakage connects the vessel to the background. (d) 3D result with shape prior has no leakage problem. (e) Final segmentation result from a different viewpoint.



(a)



(b) (c)



(d) (e)

Figure 3.6: Geometric parametric segmentation of blood vessels (taken from [FIC04]). (a) Medial axis of a branch and its elliptical circumference. (b) The initial contours of the cross-sections of blood vessels. (c) Contours after axis deformation. (c) Contours after surface deformation. (d) Final contours after axis and surface deformation.

shape of each cross-section of the vasculature. However, it has some intrinsic problems. A vasculature might not have an elliptical cross-section when it is unhealthy. Under such circumstance, geometric parametric model may not be appropriate.

3.2 Heart Segmentation Algorithms

Segmentation and reconstruction of the heart from medical images is crucial for surgical simulation. There is not a lot of research in heart segmentation. Existing algorithms for heart segmentation can be classified into three categories, i.e., intensity-based, deformable model-based and heart model-based algorithms.

The first class is mainly intensity-based, using general segmentation algorithms. In [RCR05], the 3D heart was segmented by region growing starting from selected seed points in the thresholded CT images, followed by morphological operations to break connections between different parts. The result is shown in Figure 3.7. One of the disadvantages of this algorithm is that the surface of the segmented 3D model is not smooth. It is because the algorithm is voxel-based. It does not have any constraints on the surface. And it involves much human intervention to set the seed points and the thresholds.

The second class of algorithms carries out segmentation with deformable models. Makowski et al. applied two-phase active contour to segment chambers and arteries from 2D MRI images [MST$^+$81]. Bajaj et al. used level set algorithm to segment heart from high resolution CT images [BGY$^+$06]. The algorithm starts from multiple seed points and segmented various anatomical parts simultaneously. Both algorithms handle only 2D segmentation. In 3D case, the deformable model-based algorithms become much more complicated. Forces are computed in 3D. The 3D calculation is more time-consuming. Sometimes, the equation cannot be easily extended from 2D case. Therefore, 3D deformation becomes much more difficult to

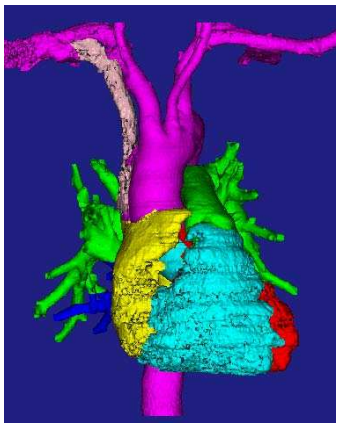


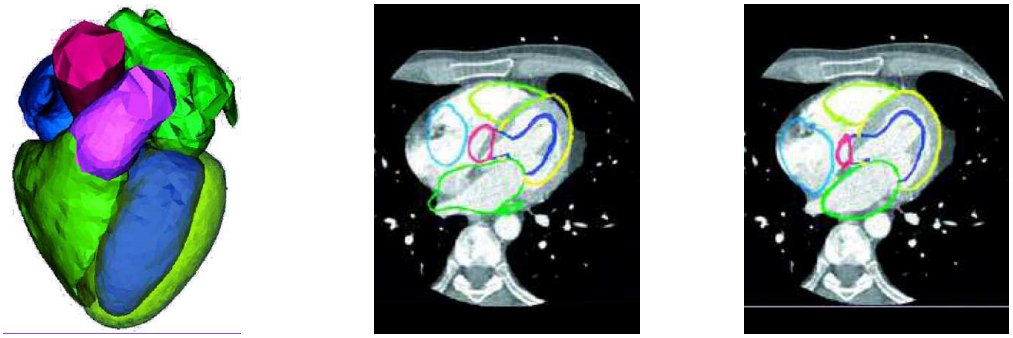
Figure 3.7: Intensity-based 3D segmentation of the heart (taken from [RCR05]).

handle than 2D case.

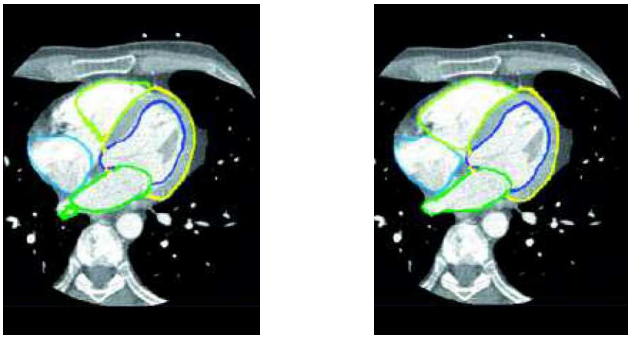
The third class of algorithms segments with a predefined heart model. It uses registration techniques to align the model to the input images. Philips Research Group proposed a segmentation algorithm with an annotated heart model [EPW+06]. The algorithm first detects the approximate position of the whole heart in CT images using generalized Hough transformation [Bal81]. Then the heart model (Figure 3.8(a)) is registered to the image, optimizing its position in space with respect to a rigid transformation (Figure 3.8(b)). After that, various anatomical regions of the heart, e.g., chambers and great arteries, are simultaneously registered to the image using piecewise affine transformation (Figure 3.8(c)). Finally, accurate boundary delineation is achieved using a mesh-deformation method (Figure 3.8(d)).

3.3 Summary

Medical image segmentation is still a big challenge. Different anatomical parts have different characteristics in different imaging modalities. No segmentation algorithm can handle all kinds of segmentation problems. Each one has its own pros and cons.



(a) (b) (c)



(d) (e)

Figure 3.8: Model-based 3D heart segmentation (taken from [EPW+06]). (a) 3D heart model. (b)-(e) Segmentation results of each stage of the method.

The vasculature and heart segmentation algorithms discussed above are all sensitive to noise to different degrees. Heart model-based algorithm is the least sensitive. When the model is available, model-based algorithm should produce the best result as it uses knowledge in segmentation. Snake and level set are more flexible for segmenting larger vessels and chambers. Snake is more sensitive to initialization than level set. Level set can handle topological changes, but snake cannot. For thin vasculature with complex topological structure, the image forces are too weak to guide the propagation of the active contour and the front of level set. In this case,

it is better to use centerline detection and geometric parametric models. Through centerline detection, topological information of the vascular structure is easy to obtain. But it is still very difficult for it to extract all the tiny vessels. Geometric parametric model-based approach does not have leakage problem because it always preserve the shape of each cross-section of the vasculature. However, this approach may have difficulties in segmenting diseased vasculature because the cross-sectional shape may not be preserved.

Chapter 4

Segmentation and Reconstruction of Arteries

4.1 Project Description

4.1.1 Description of Input Data

The input data of this project are chest CT images of newborn babies. The size of the CT images is 512×512 ((Figure 4.1)), and the inter-slice thickness is 0.625mm (Table 4.1). Each data set consists of about a hundred slices from the aortic arch to the bottom of chambers.

Data Set 1 and Data Set 2 seem to be incomplete: some parts of the great arteries are missing. This could be due to the babies' movement during CT scanning. Data Set 3 is complete.

As discussed in Chapter 2, the key to distinguishing CT images of TGA patients from normal patients is to identify the relative positions of the ascending aorta and the pulmonary trunk. In the CT images of the first and third patients (Figure 4.2(a) and (c)), the ascending aorta is located in front of the pulmonary trunk for TGA

Figure 4.1: Sample CT image.

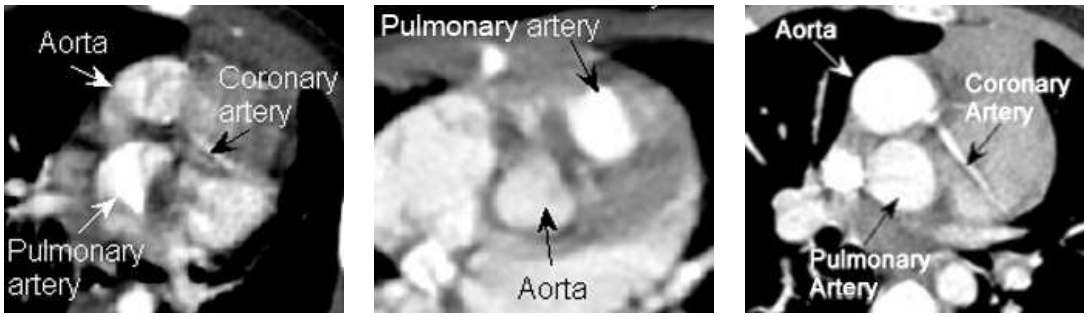
Table 4.1: Resolution of three data sets.

Data Set	$x-/y-$Resolution (mm/pixel)	Pixel size	Actual size (mm)	Thickness (mm)	TGA	Sample
No. 1	0.215	512×512	110.08 × 110.08	0.625	Yes	Figure 4.2(a)
No. 2	0.215	512×512	110.08 × 110.08	0.625	No	Figure 4.2(b)
No. 3	0.369	512×512	188.928 × 188.928	0.625	Yes	Figure 4.2(c)

cases. In the CT images of the second case (Figure 4.2(b)), the aorta is located behind the pulmonary trunk. It is the normal case.

4.1.2 Description of Problem

The objective of this project is to develop a system which can segment and reconstruct 3D artery models from CT images. Aorta is segmented aortic root to the aortic valve. Pulmonary trunk is segmented from pulmonary root to the pulmonary valve. Coronary arteries are also segmented out if they appear in the data set. The spine is also required as a reference to the orientation and posture of the patient.



(a) (b) (c)

Figure 4.2: Sample images of three data sets. (a) Data Set 1. (b) Data Set 2. (c) Data Set 3.

After segmentation, surface models of the arteries and the spine are constructed for the purpose of visualization. The models are constructed according to the resolution and thickness of CT images.

4.2 Overview of Algorithm

According to different characteristics of arteries which are discussed in Chapter 2, different semi-automatic 3D segmentation algorithms are used in this system. For the aorta and the pulmonary trunk, 3D level set is used (Section 4.3). The coronary arteries appear very thin and unclear. Their edges are vague. It is difficult to apply level set to segment them. Therefore, an alternative algorithm based on geometric models is used to segment coronary arteries (Section 4.4). The spine is segmented for visualizing the patient's orientation. The accuracy of the segmentation result is not as important as that of the great arteries. Therefore, multi-seed region growing, which is much simpler and more efficient than level set, is applied to segment the spine (Section 4.5).

After segmentation, 3D surface models of the arteries and the spine are con-

structed using marching cubes algorithm. The segmentation and reconstruction of aorta, pulmonary trunk, coronary arteries and spine may proceed in any order.

4.3 Segmentation of Great Arteries

4.3.1 Overview

Two-phase 3D level set algorithm is used to segment the aorta and the pulmonary trunk. The first phase is to get an approximate surface of the great artery by using gradient map of CT images. The gradient information is not enough to obtain an accurate segmentation result due to the inhomogeneity of the images. Detailed reason and analysis will be covered later. A second phase is applied to expand the surface a little using region-based information to obtain more accurate surface.

4.3.2 Level Set with Gradient Map

The main idea of level set algorithm is discussed in Section 3.1.3. Level set function $\phi(x, y, z)$ is applied in 4D. The rate of change $\partial\phi/\partial t$ of ϕ is related to the gradient $\nabla\phi$ as follows:

$$\frac{\partial\phi}{\partial t} + F|\nabla\phi| = 0. \tag{4.1}$$

Usually the force F is formulated as $F = F_P + F_G$. The term F_P, referred to as the propagation term, controls the propagation of the front, and stops the front in the vicinity of the desired boundaries. Usually the term is computed from the input images.

The term F_G depends on the geometry of the front. It aims at smoothing out the high curvature regions of the front. The great arteries usually appear elliptical in CT images. Their outer surfaces should be smooth. High intensity noise in the

CT images may attract the front wrongly. It will result in rough surface. The curvature is usually relatively high where it is rough. To avoid that, F_G is inversely proportional to the curvature $\kappa(x)$. In regions of high curvature, it will decrease the speed of the front. Therefore, the front in the high curvature regions will propagate slower than that in the low curvature regions. As the front moves in the normal direction, the high curvature regions tend to be smoothed out.

In this algorithm, the level set function is rewritten as follows:

$$\begin{aligned} F_P &= a\, v_I(x) \\ F_G &= -b\, v_I(x)\, \kappa(x) \\ F &= F_P + F_G = v_I(x)\,[a - b\,\kappa(x)] \\ \frac{\partial \phi}{\partial t} &+ v_I(x)\,[a - b\,\kappa(x)]\,|\nabla \phi| = 0, \end{aligned} \tag{4.2}$$

where a and b are nonnegative scalar constants, weighting the relative influence of each term on the movement of the front. $v_I(x)$ is defined as a speed function based on the gradient map of the input images. As required, the front should stop near the artery boundaries. The value of the speed function should decay to zero when the front is approaching the boundaries.

There are three different formulae for the speed function $v_I(x)$.

$$v_I(x) = (1 + |\nabla I(x)|)^{-1} \tag{4.3}$$

$$v_I(x) = \exp\left(-\gamma |\nabla I(x)|\right) \tag{4.4}$$

$$v_I(x) = (1 + \exp\left(-\frac{|\nabla I(x)| - \beta}{\alpha}\right))^{-1}, \tag{4.5}$$

where $|\nabla I|$ is the gradient magnitude of input image I. I should be Gaussian smoothed to remove noise.

In comparison, the advantage of exponential function (Equation 4.4) over inverse proportion function (Equation 4.3) is that exponential function decays more quickly. Equation 4.5 uses sigmoid function. α, β are two parameters required by sigmoid

function to define the nonlinear transformation. The sigmoid function enlarges the speed difference of the front between in regions of low gradient magnitude and regions of high gradient magnitude. The speed of convergence of sigmoid function is adjustable and controlled by α, β.

Ideally, the speed function should have non-zero value in the homogeneous regions of the great arteries. The front can propagate inside the great arteries. The speed function should decay rapidly to zero near their boundaries. Otherwise, the front will not stop at the boundaries if the speed is nonzero. According to the three formulae of $v_I(x)$, the value cannot equal to zero unless the gradient magnitude is infinite, which is obviously impossible in real applications. Therefore, a threshold is needed.

$$v_I'(x) = \begin{cases} v_I(x) & \text{if } v_I(x) > \Gamma, \\ 0 & \text{otherwise.} \end{cases} \tag{4.6}$$

The threshold Γ should be large enough. Otherwise, the front will leak out because the speed function still has nonzero values somewhere at the boundaries. On the other hand, if Γ is too large, the front will stop before it arrives at the actual boundaries. It is observed that areas inside the great arteries appear inhomogeneous in CT images. Regions around the arteries also have different intensities. Therefore, the gradient magnitudes of the voxels on the boundaries vary quite a lot. In this case, it is very hard to select a perfect value for the threshold Γ.

As discussed before, one of the main disadvantages of level set is its tendency to leak out. Thus, threshold Γ should be set to a stricter value to avoid leaking. It should be larger than the speed values of all the voxels at the boundaries. Setting Γ in this way causes problem that the front may stop before it gets to the actual boundaries. Figure 4.3(b) shows the speed functions $v_I'(x)$ after thresholding. Pixels on the line (Figure 4.3(a)) are chosen as sample points to show the change of the speed when the front propagates along this line. The high gradient magnitude

indicates the positions of boundary points. However, due to the value of threshold Γ, $v_I'(x)$ decays to zero before it gets to the boundary points.

In comparison, the curve of sigmoid function is more smooth than the others. Additionally, the parameters α and β of sigmoid function can control the non-zero range. It is more flexible than the other functions. Therefore, it is chosen as the speed function in the segmentation algorithm.

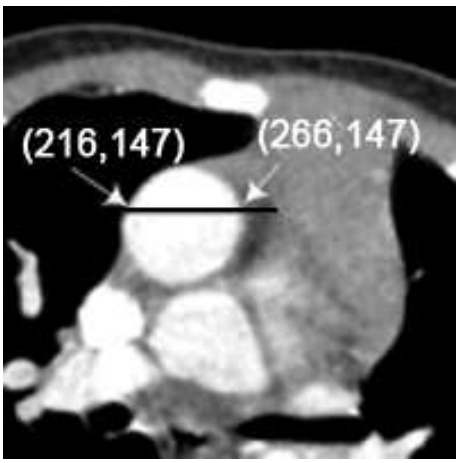
The values of α, β are found in the following way suggested by ITK's documentation [ITK]. Let K_1 be the minimum gradient magnitude on the contour of the great arteries. Let K_2 be the average value of the gradient magnitude in the middle of great arteries. The two values indicate the dynamic range that needs to be mapped to the interval $[0, 1]$ in the speed function. Therefore, the suggested value for β is $(K_1 + K_2)/2$, and the value for α is $(K_2 - K_1)/6$ which is negative.

The algorithm requires the user to specify the initial surface for level set. It is a sphere inside the desired artery. Sample segmentation results of the aorta and the pulmonary trunk are shown in Figure 4.4. The results indicate that level set guided by gradient map cannot get the accurate surface of the desired arteries. It gets stunk at a few pixels away. Therefore, it is necessary to apply another algorithm to get more accurate result.

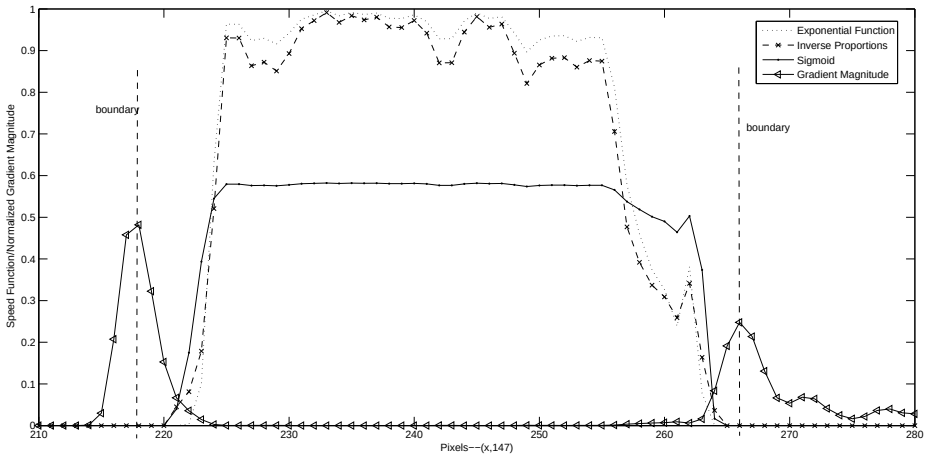
4.3.3 Masking

The aim of masking is to pre-process original CT images as the input for the second phase according to the segmentation result of the first phase. The purpose of masking is to restrict the computational field of level set algorithm and avoid negative influence of irrelevant tissues in the CT images. The main idea is to mask out regions far away from the desired arteries in the input CT images.

Masking includes two steps. The first is to expand the surface obtained in the

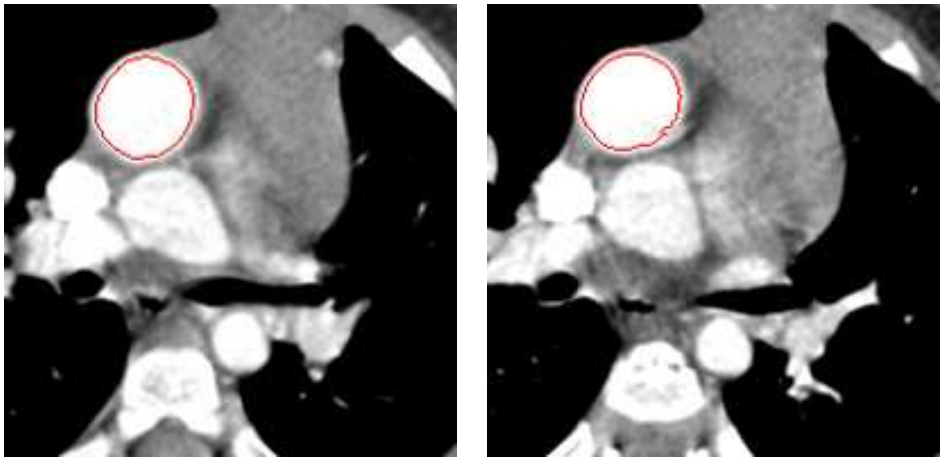


(a)



(b)

Figure 4.3: Comparisons of the speed functions. (a) Sample pixels chosen along the black line. (b) Comparison among speed functions and gradient magnitudes along the black line.



(a) (b)

Figure 4.4: Segmentation results of level set in the first phase.

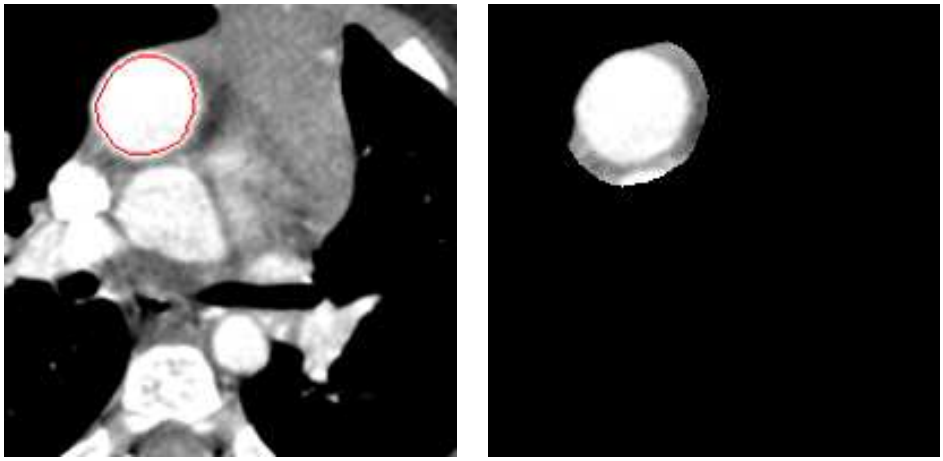
first phase by a predefined amount d_m. This is achieved by computing the minimal distances of the voxels to the surface. The expanded volume consists of all the voxels with distances $d \leq d_m$. The second step is to remove the regions outside the expanded volume. Note that d_m must be large enough to include all the parts of the great arteries.

Figure 4.5(a) shows the segmentation result of the first phase. Figure 4.5(b) is obtained with $d_m = 10$ pixels.

4.3.4 Level Set with Region Based Information

The objective of this step is to move the level set front a little further to let it stop at the desired boundaries. The algorithm is based on the paper of Dindoyal et al. [DLD+05]. The idea is to model the foreground (e.g., great arteries) and the background of the image and try to minimize the energy of separating these two regions. The level set function is as follows:

$$\frac{\partial \phi}{\partial t} + [\lambda_1 (I - \mu_o)^2 - \lambda_2 (I - \mu_i)^2] \exp(\omega \kappa) |\nabla \phi| = 0. \tag{4.7}$$



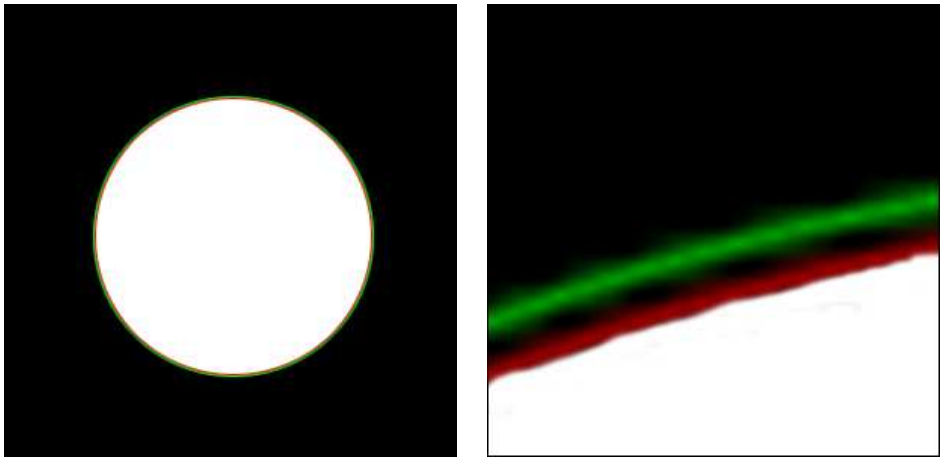
(a) (b)

Figure 4.5: Result of masking. (a) Segmentation result of the first-phase level set algorithm. (b) Masked image.

In Equation 4.7, I is the voxel intensity. μ_i and μ_o are average intensities inside and outside the region defined by the level set front. μ_i and μ_o should be estimated iteratively. λ_1 and λ_2 are empirically determined weighting coefficients. Since the appropriate λ_1 and λ_2 could potentially vary significantly between data sets, the input images are normalized by mapping intensities between 0 and 1 linearly.

The region-based force $[\lambda_1(I-\mu_o)^2 - \lambda_2(I-\mu_i)^2]$ mainly depends on the local tissue type. It will also work in the absence of a strong edge field because it is not based on input images' gradient. The force is heavily penalized by curvature κ to prevent leakage of the front.

Let us examine the region-based force using the image shown in Figure 4.6. The white part has intensity 1 and the black part has intensity 0. Assume that $\lambda_1 = \lambda_2 = 1$. When the front is inside the white region, $\mu_i = 1$, $0 < \mu_o < 1$ and the intensities of the pixels at the front equal to 1. At this time, the region-based force is positive. The front will expand. When the front is outside the white region. $0 < \mu_i < 1$, $\mu_o = 0$, and the intensities of the pixels at the front equal to 0. Now the

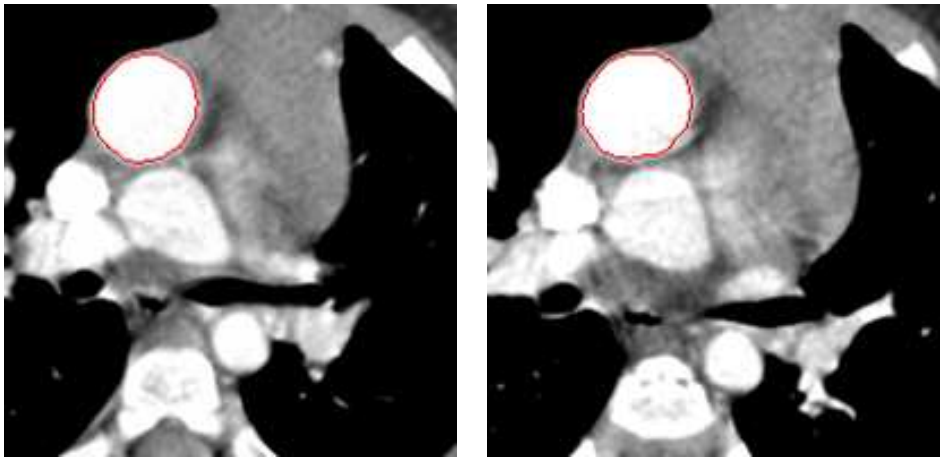


(a) (b)

Figure 4.6: Image to examine the region-based force. (a) A binary image for segmentation. (b) Zoom in view.

force is negative, and the front will shrink. Therefore, the front will stop between the boundary with intensity equal to 1 (red contour in Figure 4.6(b)) and the boundary with intensity equal to 0 (green contour in Figure 4.6(b)). In this project, level set algorithm has not taken the subpixel problem into consideration. It will not stop the front in the middle of the red and the green contours. The red contour will be chosen as the boundary.

The initial surface is the surface obtained in the first phase (Figure 4.4). The masked CT images are the input images. The front propagates in 3D under the influence of the region-based force. Figure 4.7 shows the result after 78 iterations. In comparison with Figure 4.4, now the front stays much closer to the real boundaries of the aorta.



(a) (b)

Figure 4.7: Segmentation results of the second phase level set.

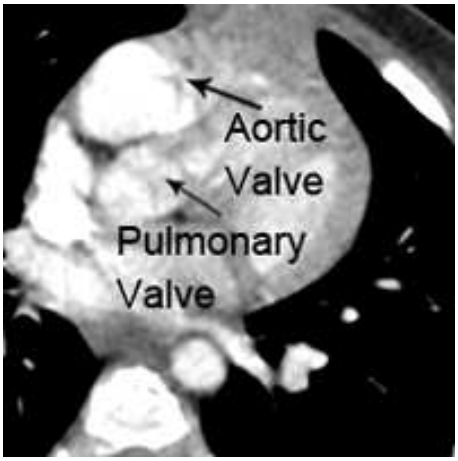


Figure 4.8: Aortic valve and Pulmonary valve of Data Set 3.

4.3.5 Experiment and Discussion

The segmentation algorithm of the great arteries was tested on the three data sets described in Section 4.1.1. As required, aorta was segmented from aortic arch to aortic valve, and pulmonary trunk was segmented from pulmonary root to the pulmonary valve ((Figure 4.8)). Data Set 3 has a higher quality than Data Set 1 and 2. So, the segmentation algorithm was applied to more CT slices to obtain more complete artery model, including the aortic arch and the pulmonary trunk.

There were three steps in the segmentation of an artery. First, the first-phase level set algorithm was run on the input images with a manually specified initial surface inside the desired artery. Secondly, the input images were masked according to the result of the first phase. Thirdly, the second-phase level set algorithm was applied to the masked images. These steps were repeated to segment the aorta and pulmonary trunk in the three data sets.

The algorithms are implemented using Medicine Insight Segmentation and Registration Toolkit (ITK) [ITK]. It provides the data structure for reading, writing and storing 2D and 3D images. It also provides a framework for the level set algorithm. Users need to specify the computation of speed function, but do not need to worry about solving partial differential equation.

The parameters of the algorithms were set as shown in Table 4.2. a and b are weighting factors that balance the influences of propagation term and geometric term. They are set to 3 and 0.5 so that the front can expand to the vicinity of the boundaries and it also can remain smooth. α and β of the sigmoid function were estimated from input images. The guideline for setting α and β is discussed in Section 4.3.2. The threshold Γ of the speed function was set to 0.002 for all the data sets so that the front can stop expanding when the speed is close to 0. Sigmoid speed function is controlled by α and β. So it is less dependent on the value of Γ.

Table 4.2: Parameter configuration for algorithms. A: aorta, P: pulmonary trunk, No.: number of iterations.

Data Set	Arteries	First Phase						Masking	Second Phase		
		a	b	α	β	Γ	No.	d_m	λ_1	λ_2	No.
1	A	3	0.5	-6	2	0.002	400	10	0.3	0.3	100
	P	3	0.5	-5	1	0.002	800	10	0.3	0.3	100
2	A	3	0.5	-5.5	2	0.002	400	15	0.3	0.3	200
	P	3	0.5	-5.5	2	0.002	400	15	0.3	0.3	200
3	A	3	0.5	-2	1	0.002	1500	20	0.3	0.3	500
	P	3	0.5	-4	1	0.002	800	20	0.3	0.3	500

The expanding distance d_m for masking should be larger than the maximal distance between the surface obtained by the first phase and the real surface of the artery. λ_1 and λ_2 are the weighting parameters in the second phase. They are set to 0.3 so that the front can expand slowly with a more precise result. The numbers of iterations for both the phases depend on the volume of the artery and the propagation speed of the front.

Figure 4.9, 4.10 and 4.11 show sample segmentation results of the data sets. The boundaries in Data Set 3 are clearer and sharper than those in Data Set 1 and 2. Its gradient magnitude map is steeper at the boundaries. Therefore, the surface obtained in the first phase ((Figure 4.11) is nearer to the actual boundaries than those in other data sets (Figure 4.9 and 4.10).

Although the results of the first phase are not close enough to the actual boundaries, region-based level set algorithm in the second phase can compensate for the shortcomings. As shown in Figure 4.12(b), segmentation results obtained in the second phase are satisfactory expect for a few slices with high image noise (e.g.,

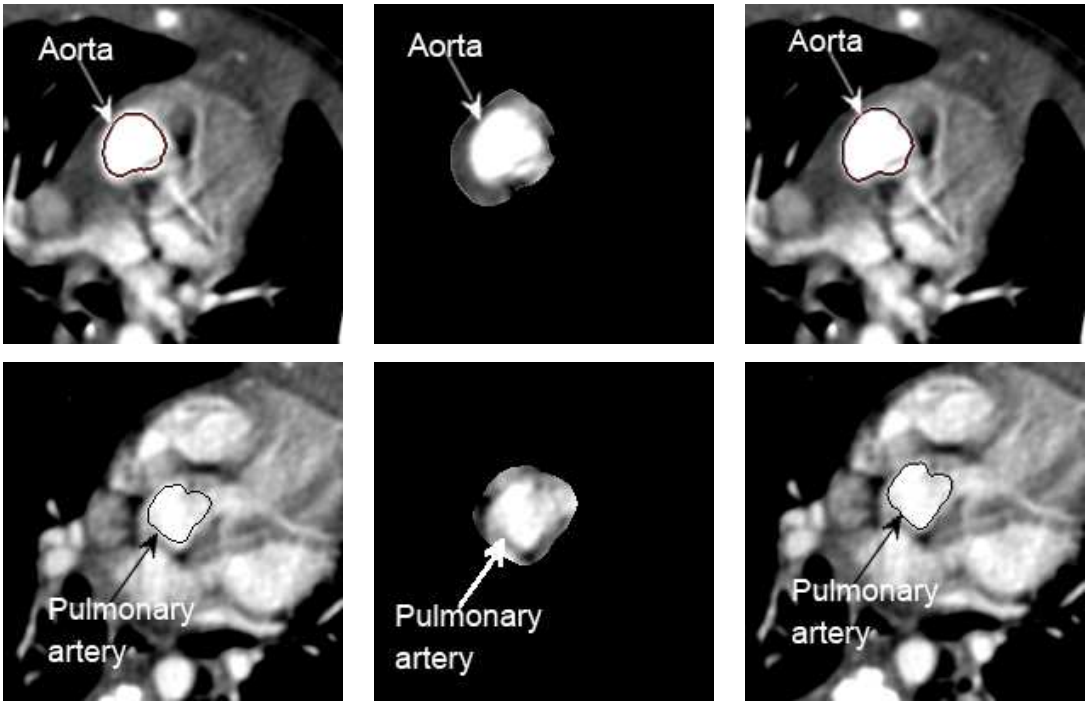


Figure 4.9: Sample segmentation results of Data Set 1. First column: results of the first phase. Second column: masked input images. Third column: results of the second phase.

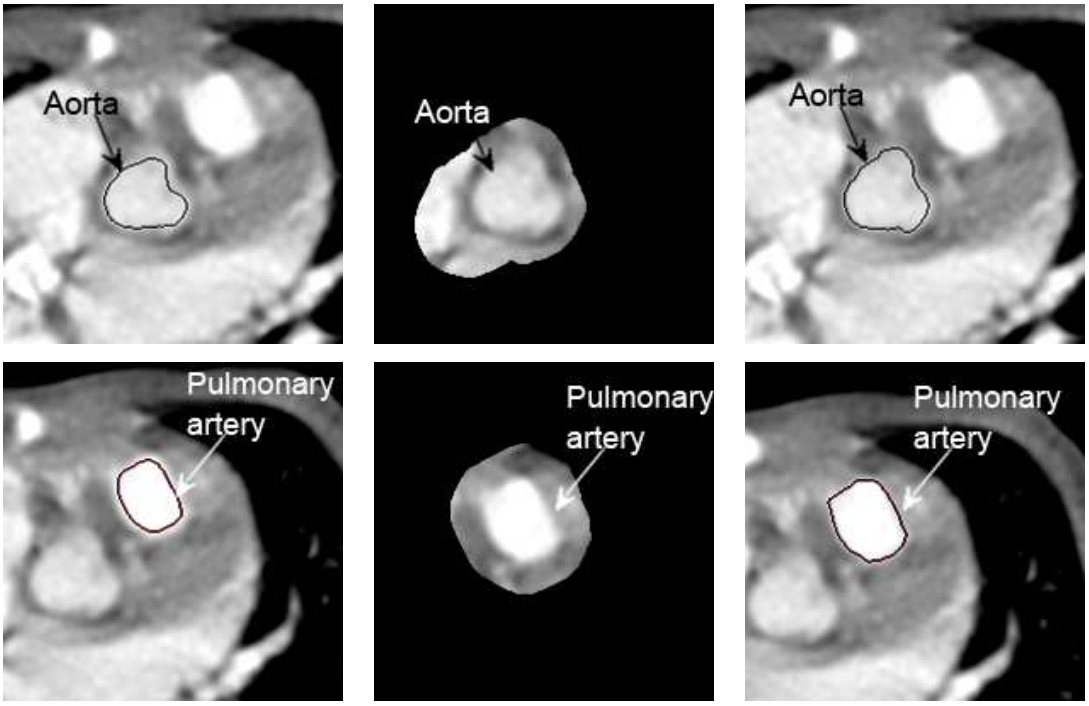


Figure 4.10: Sample segmentation results of Data Set 2. First column: results of the first phase. Second column: masked input images. Third column: results of the second phase.

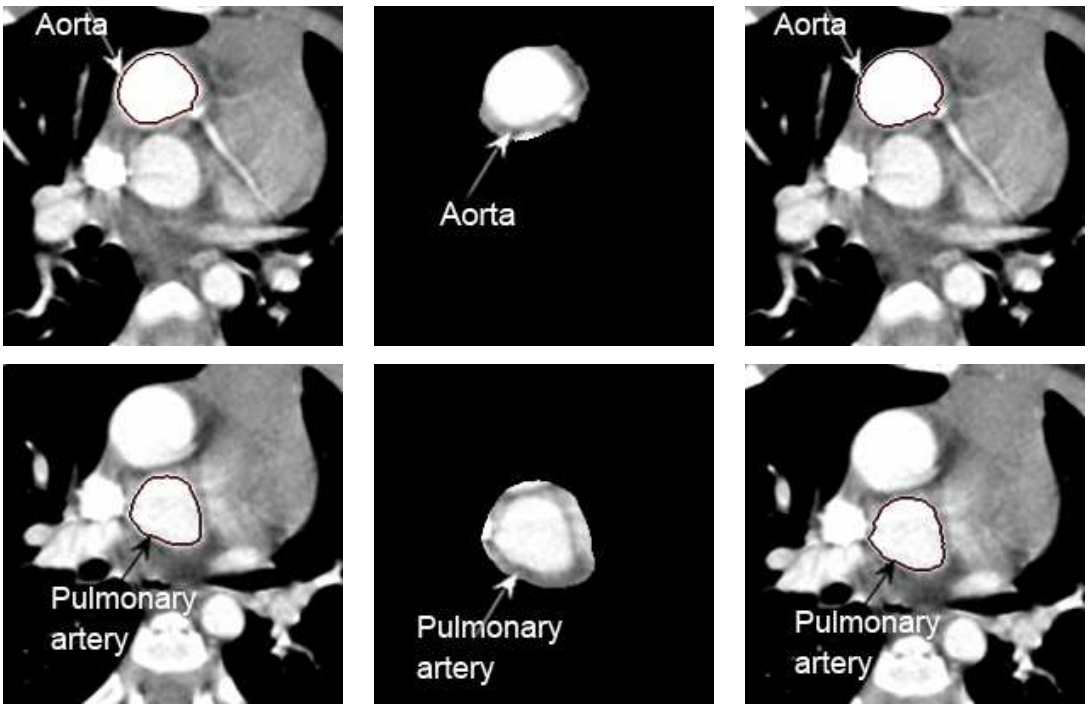
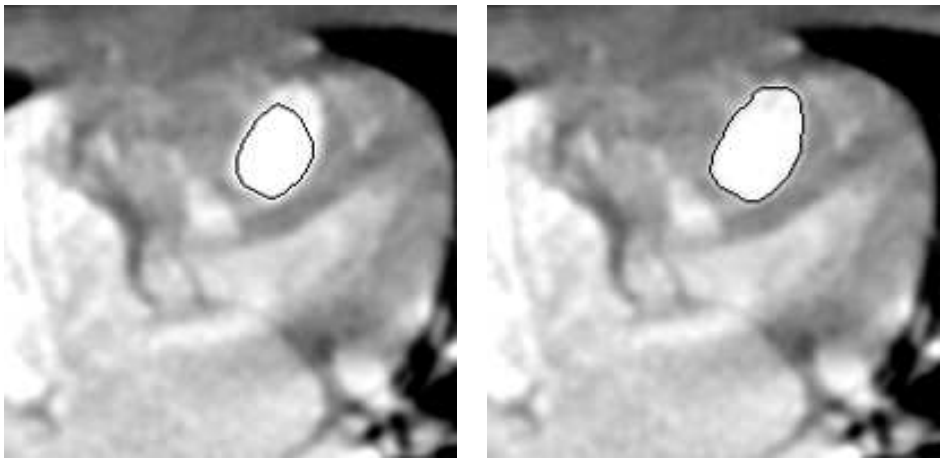
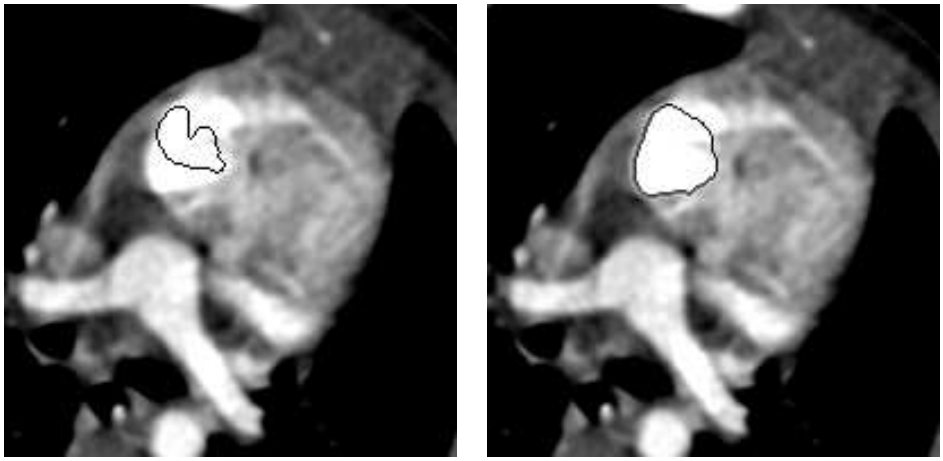


Figure 4.11: Sample segmentation results of Data Set 3. First column: results of the first phase. Second column: masked input images. Third column: results of the second phase.



(a) (b)

Figure 4.12: Segmentation results of Data Set 2. (a) Result of Phase 1. (b) Result of Phase 2.



(a) (b)

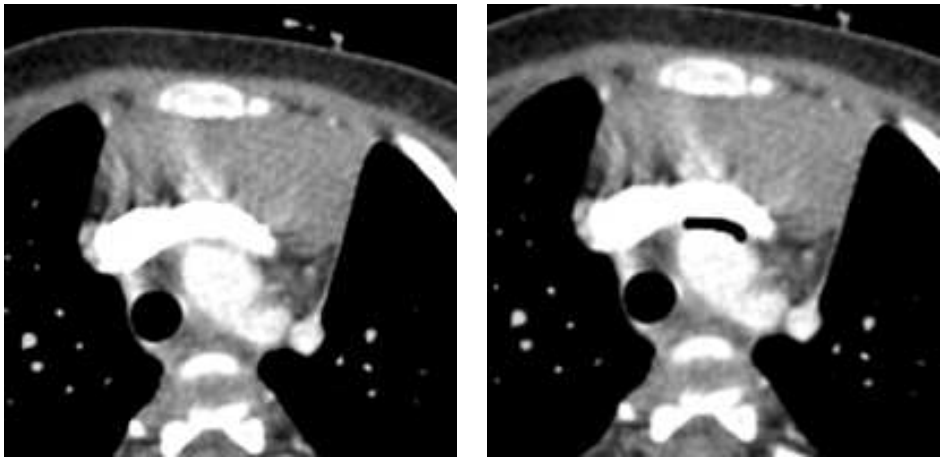
Figure 4.13: One unsatisfactory segmentation result of Data Set 1. (a) Result of Phase 1. (b) Result of Phase 2.

Figure 4.13).

Since Data Set 3 is of high quality, the aortic arch and pulmonary trunk are also segmented to build complete cardiac artery models. In some slices, the boundaries between the aortic arch and the other vessels are not clear, the pixels between them have high intensities, which usually confuses the level set algorithm and results in leakage. In that case, level set algorithms will definitely leak out to the undesired area. So before segmentation, the CT images needs to be manually pre-processed by putting black lines on unclear boundaries (Figure 4.14). Figure 4.15 shows the segmentation results of the top parts of great arteries.

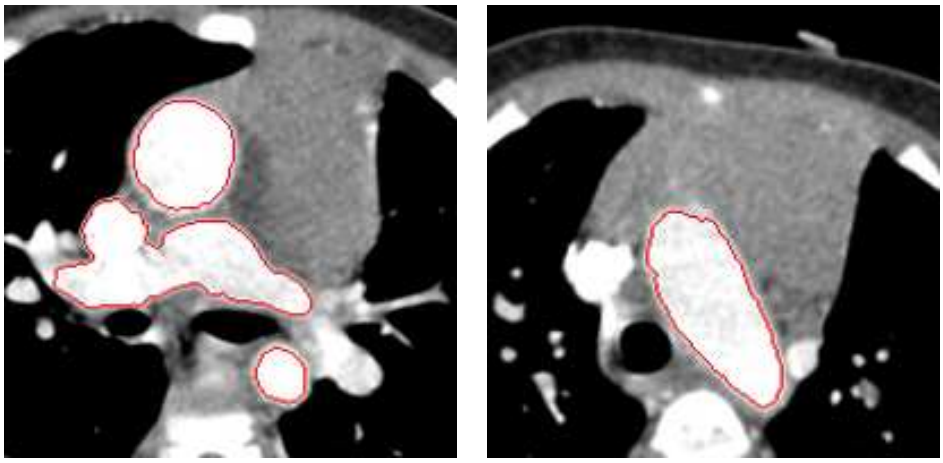
4.4 Segmentation of Coronary Arteries

Coronary arteries are very narrow. They show up in at most 4 to 5 slices. It is quite difficult to reconstruct an accurate 3D model of coronary arteries. So some estimation needs to be performed. Due to the shape of the coronary arteries,



(a) (b)

Figure 4.14: Sample image after manual pre-processing. (a) Original CT image, (b) image after separating vessels manually.



(a) (b)

Figure 4.15: Segmentation results of top parts of great arteries. (a) Ascending aorta, descending aorta and pulmonary trunk. (b) Aortic arch.

their centerlines are very important for identifying their positions and orientations. Therefore, the first step is to determine the centerlines of the coronary arteries. The second step fits geometric parametric models to the intersections and centerlines of coronary arteries in the CT images, and obtains the outer surface of the coronary arteries.

4.4.1 Centerline Detection

As discussed in Section 3.1.1, centerline detection algorithms are designed to find the centerlines of the entire vasculature. They are all based on the assumption that centerlines are the brightest lines along the blood vessels.

Here, a centerline detection algorithm based on graph theory is applied to the 3D data. It requires the user to indicate two end points of the coronary arteries. They do not have to be in the same CT slice. The main idea is to find the minimum-cost path between the two end points.

First, a directed graph G is built from the 3D volume data. Each vertex in G corresponds to a voxel in the 3D volume data. Let V_i be a voxel in the 3D data. There are three categories of adjacent voxels of V_i (Figure 4.16):

1. 6 voxels share a face with V_i,
2. 12 voxels share an edge with V_i,
3. 8 voxels share a corner with V_i.

V_i is adjacent to these 26 voxels in 3D data. The distances between them are 1, $\sqrt{2}$, and $\sqrt{3}$ respectively.

The cost $c(i, j)$ of an edge (V_i, V_j) is defined as

$$c(i, j) = w_1(1 - I_j) + w_2 d(i, j). \tag{4.8}$$

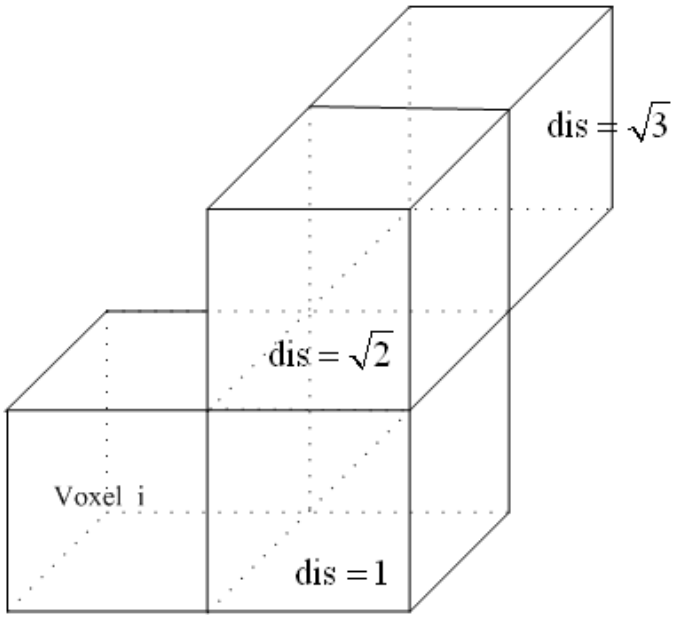


Figure 4.16: Distance between adjacent voxels in 3D space.

In this equation, the first part of the cost is related to intensity. Based on the assumption that centerlines are the brightest lines along the coronary arteries, the path should go through high intensity points that have higher possibility of belonging to the centerline. I is the normalized intensities of input images with a maximum value of 1. The other part $d(i, j)$ is the distance between V_i and V_j. Sometimes, there is some noise with high intensity around the coronary arteries. The path through this kind of noise vertexes may have low intensity cost. But it usually has large distance cost because it deviates the coronary arteries' centerline. Distance constraint is set to reduce the influence of noise. The parameters w_1 and w_2 are two constants set by the user. Given the end points, the algorithm uses Dijkstra's algorithm [CLRS01] to find the minimum cost path. Figure 4.17 shows the centerline by projecting to one slice.

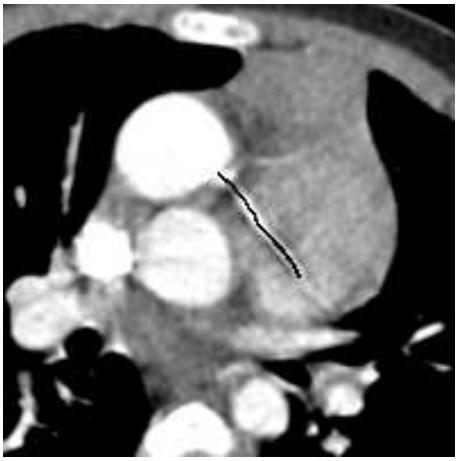


Figure 4.17: Detected centerline in one slice.

4.4.2 Fitting Geometric Parametric Model

After getting the centerline, a geometric parametric model is fitted the surface of the coronary arteries. Suppose that the centerline L_c is composed of a set P of points. For each point $p_i(x_i, y_i, z_i)$ in P, a sphere is placed at p_i to fit the intersection of the coronary arteries in input images. A sphere has 4 parameters, the coordinates of the center (x_c, y_c, z_c) and the radius r. The parameters of the sphere are estimated by minimizing the cost function

$$C = a_1 C_i + a_2 C_r + a_3 C_d. \tag{4.9}$$

The first part is intensity. The idea is to find an optimal sphere that can classify the voxels into two groups with minimum intensity difference. The formula is written as

$$\begin{aligned} C_{\mathrm{i}} &= \sum_{\mathrm{d}(p',p_i)\le r} |I(p') - I_{\mathrm{i}}| + \sum_{r\le \mathrm{d}(p'',p_i)\le 2r} |I(p'') - I_{\mathrm{o}}| \qquad (4.10) \\ d(p', p_i) &= \sqrt{(x'-x_i)^2 + (y'-y_i)^2 + (z'-z_i)^2}, \end{aligned}$$

where I_i is the average intensity of voxels inside the coronary artery, and I_o is the average of those outside the coronary artery. The values of I_i and I_o are computed

before fitting. They are determined by examining sample voxels given by users inside and outside the coronary arteries before segmentation. Define region inside a sphere (x_c, y_c, z_c, r) as the foreground and region outside the sphere (x_c, y_c, z_c, r) but within the sphere $(x_c, y_c, z_c, 2r)$ as the background. Voxels beyond these regions are not taken into consideration. The first term in Equation 4.10 sums over the intensity differences between I_i and foreground voxels. The second term sums over the intensity differences between I_o and background voxels.

The second part of the cost function is related to radius of sphere. As the segmented coronary arteries are supposed to be as smooth as possible, the difference in radius between two adjacent spheres should not be too large. The cost related to radius is defined as

$$C_r = |r - r'|, \tag{4.11}$$

where r' is the radius of adjacent sphere.

The third part imposes constraints on the distance between the sphere's center and the centerline. The center of the sphere will be searched only in the vicinity of each point on the centerline.

$$C_{\mathrm{d}} = \mathrm{d}(p_c, p_i), \tag{4.12}$$

where p_c is the center of the sphere. Its coordinates are (x_c, y_c, z_c). p_i is the point on the centerline under investigation. Its coordinates are (x_i, y_i, z_i).

The final form of cost function is the combination of these three parts with three weights.

$$\begin{aligned} C &= a_1 C_{\mathrm{i}} + a_2 C_r + a_3 C_{\mathrm{d}} \\ &= a_1 [\sum_{\mathrm{d}(p', p_i) \le r} |I(p') - I_{\mathrm{i}}| + \sum_{r \le \mathrm{d}(p'', p_i) \le 2r} |I(p'') - I_{\mathrm{o}}|] \\ &+ a_2 |r - r'| + a_3 \mathrm{d}(p_c, p_i), \end{aligned} \tag{4.13}$$

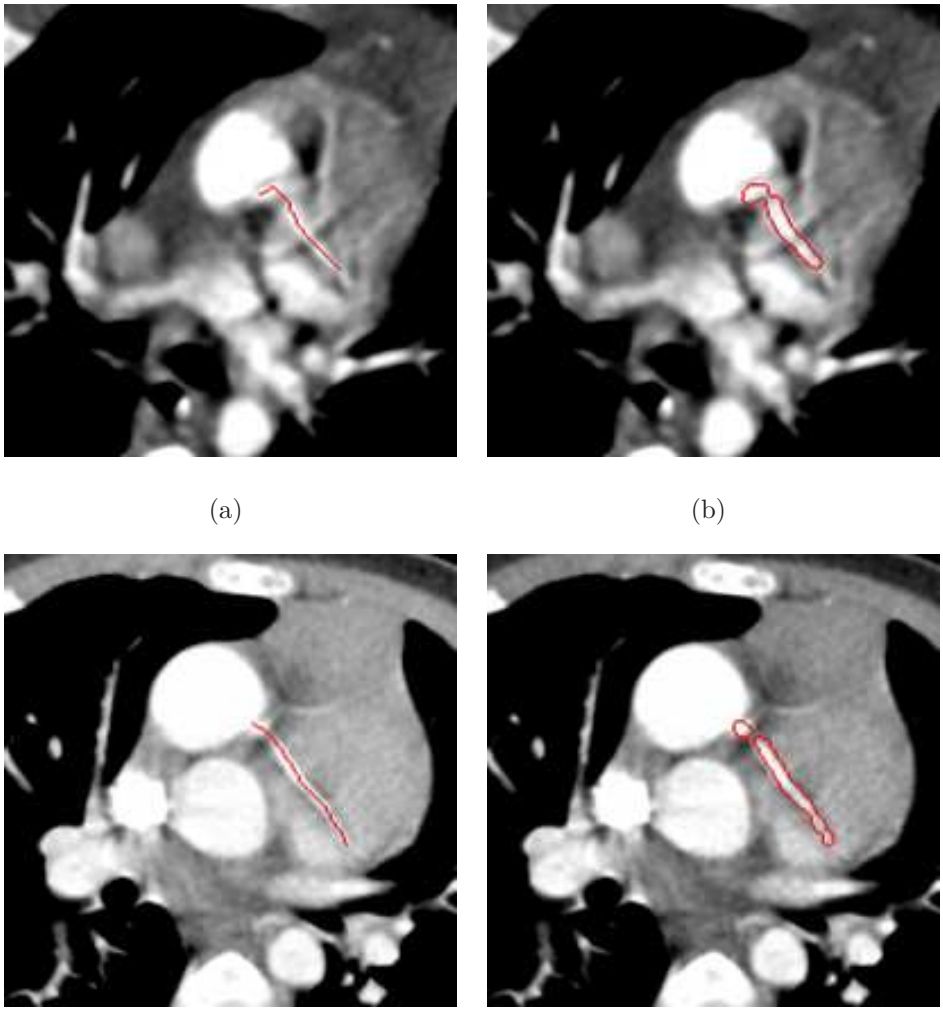
where a_1, a_2, and a_3 are coefficients set by the user. For each point on the minimal cost path, the fitting cost is minimized by determining the parameters $p_i(x_c, y_c, z_c, r)$ of the sphere. The space covered by these spheres are regarded as coronary arteries.

4.4.3 Experimental Result

Segmentation algorithm of coronary arteries is tested on Data Set 1 and Data Set 3. CT images in Data Set 2 are a bit blur. It is very difficult to distinguish coronary arteries from other tissues on the images. Therefore, Data Set 2 is not used in the test.

The implementation of cost function optimization is based on the numerical library vnl [vnl] which provides Nelder-Meade downhill simplex algorithm [NM64]. Coefficients a_1, a_2 and a_3 are set to 1, 0.1 and 0.1 so that the algorithm can extract the line in the center and avoid the influence of noise. Figure 4.18 shows the results of coronary artery segmentation of Data Set 1 and Data Set 2. The images in the first row show the centerline of coronary arteries by projecting to one slice. Detected centerlines are more or less located in the middle of the coronary arteries. They are almost not affected by noise. The images in the second row show the segmentation results of some slices, in which the coronary arteries are the thickest. The results are very close to the actual boundaries except for the end of the coronary arteries because they are extremely thin and blur at the end.

The main problem with this algorithm is it could not produce a result with very smooth boundaries. The first reason is CT images do not have high enough resolution for coronary arteries. As discussed, coronary arteries are very thin and only show up in three or four slices. The algorithm does not perform subpixel localization. Therefore, algorithm possibly needs to segment with subpixel accuaracy. The other reason is that constraining only on the radius of adjacent spheres is probably



(a) (b)

(c) (d)

Figure 4.18: Results of centerline detection and coronary arteries segmentation. (a,b) Data Set 1. (c,d) Data Set 3.

not enough to produce smooth boundaries.

4.5 Segmentation of Spine

The spine is segmented to show the orientation and posture of the patient in 3D visualization. The segmentation algorithm should obtain a result with correct position and approximate shape, but not necessarily accurate. Therefore, 3D region growing which is a general segmentation algorithm is applied to segment the spine.

The main idea of 3D region growing is to look for groups of voxels with similar properties. It starts from one or more seed points that serve as the initial regions. Then, it adds neighboring voxels one at a time into the regions if they have similar properties with respect to predefined criteria.

The key problem in spine segmentation using region growing is to define proper similarity criteria. Like the great arteries, the spine also has high intensity in CT images (Figure 4.19(a) and (b)). Therefore, the easiest way is to apply thresholding technique for segmentation. But in practice, it is very difficult to find a proper value of the threshold that works for the whole data set. The lower bound of the intensity the spine varies a lot among different slices. On the other hand, some tissues around the spine have higher intensity than some parts inside. Therefore, if the threshold is too low, undesired tissues will be included in the segmentation result. Otherwise, the algorithm may not obtain the entire spine.

As illustrated in Figure 4.19(a) and (b), the boundaries between the spine and other tissue are quite clear most of the time. That means the gradient information will be useful for segmenting the spine. Figure 4.19(c) and (d) show the gradient maps of Figure 4.19(a) and (b). Then, region growing algorithm starts from multiple seed points that are specified by users, and looks for neighboring voxels which have low gradient magnitudes (Figure 4.19(e,f)). Due to the thick boundaries in

gradient map, the boundaries obtain by region growing may not coincide exactly the actual boundaries. This situation is similar to that in the first phase of great artery segmentation. As discussed before, segmentation result of the spine is not required to be very accurate. Therefore, the boundaries are expanded by 2 to 3 voxels in the normal direction of boundaries obtained by region growing to obtain a more realistic result (Figure 4.19(g,h)).

Figure 4.20 shows some results of spine segmentation of three data sets. They are not necessarily accurate at each voxel. But, the positions are correct, and the shape of the spine is correct.

4.6 Reconstruction

4.6.1 Marching Cubes Algorithm

The input and the output of the segmentation algorithms discussed above are both volume data. In order to visualize the surface of the arteries, a surface model need to be constructed.

Marching cubes [LC87] is a computer graphics algorithm for extracting a polygonal mesh from volume data. It computes iso-surfaces from the discrete volume data, and produces the triangle mesh by connecting the patches from all the cubes on the iso-surface boundaries. The input data is the segmented volume data of the aorta, the pulmonary trunk, the coronary arteries and the spine. The outputs are is triangular mesh models. The mesh models represent the surfaces by vertexes and triangles that connect the vertexes.

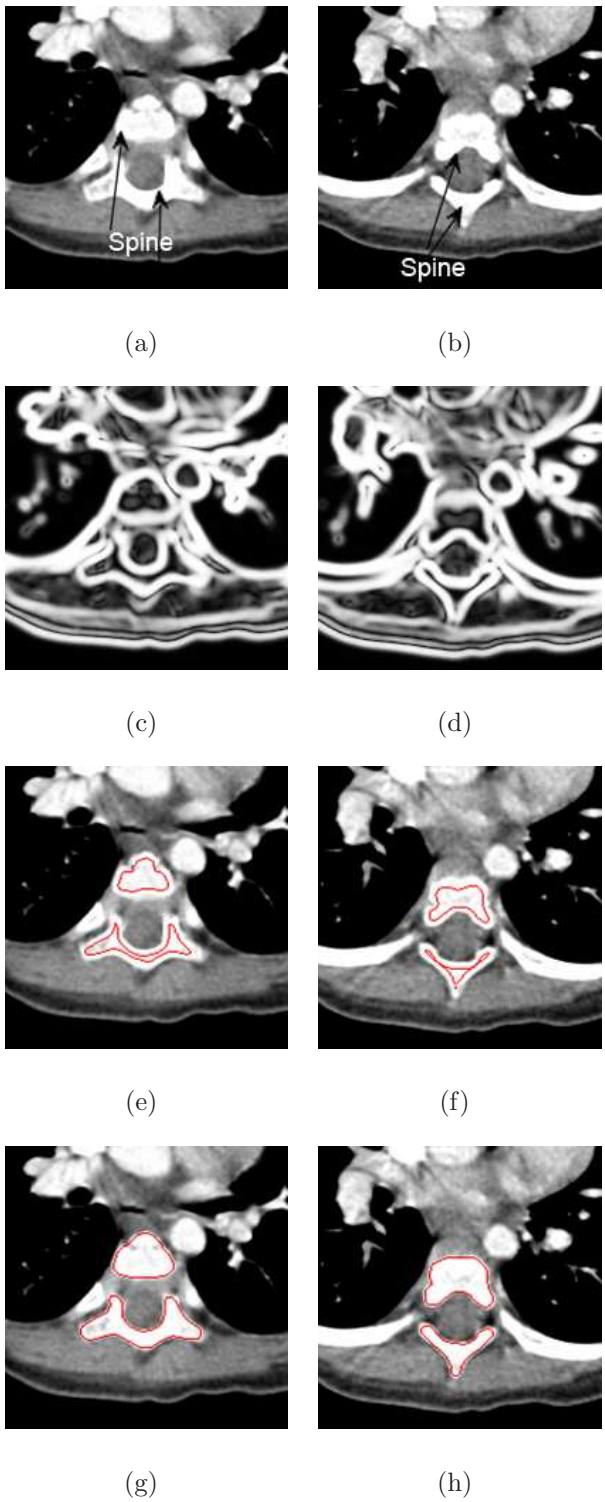
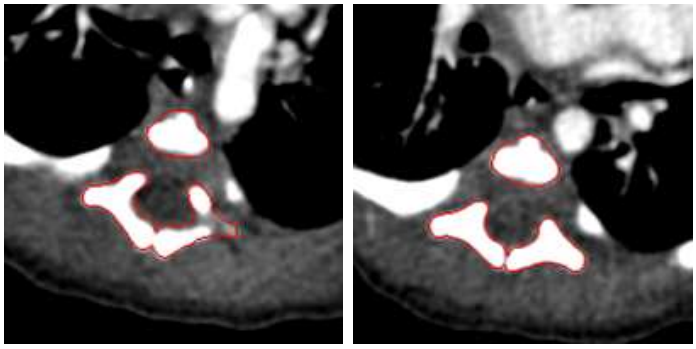
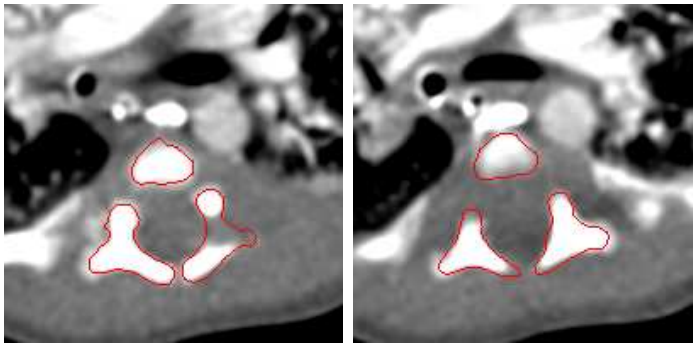


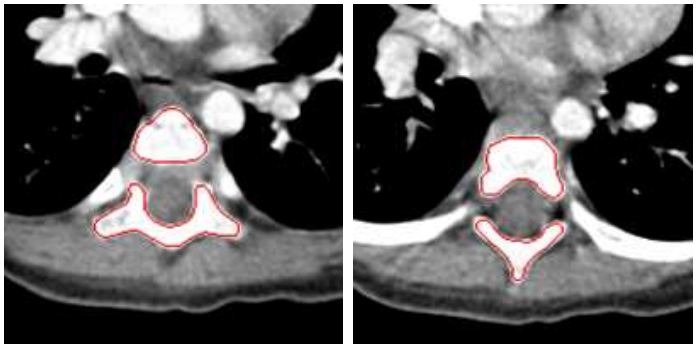
Figure 4.19: Intermediate results of spine segmentation. (a,b) Input CT images, (c,d) gradient maps, (e,f) results obtained by region growing, and (g,h) results after expansion.



(a)



(b)



(c)

Figure 4.20: Segmentation results of the spine. (a) Data set 1. (b) Data Set 2. (c) Data Set 3.

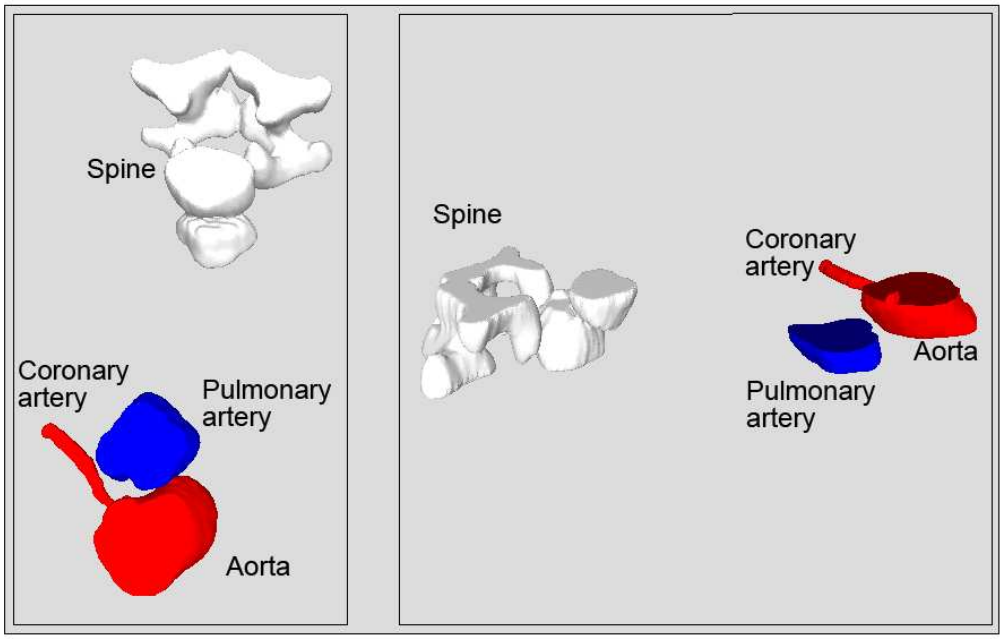


Figure 4.21: Screenshots of 3D reconstruction of Data Set 1.

4.6.2 Experimental Result

Marching cubes algorithm was tested on the segmented volume data of each artery and the spine respectively. It was implemented using ITK library [ITK]. The surface model generated by marching cubes was then visualized in a 3D visualization software after some smoothing.

Screenshots are taken to show the 3D visualization for each data set (Figure 4.21, 4.22, and 4.23). In the figures, the aorta and the coronary arteries are shaded red, the pulmonary trunk blue, and the spine white. The spine helps to determine the front and back of the patient. The spine includes two parts. The body of vertebra, more or less cylindrical in shape, faces the front of the human body. The vertebral arch is located behind the vertebral body. The sealed ends of the great arteries indicate the approximate positions of the aortic valve and pulmonary valve. The heart is located below the valves.

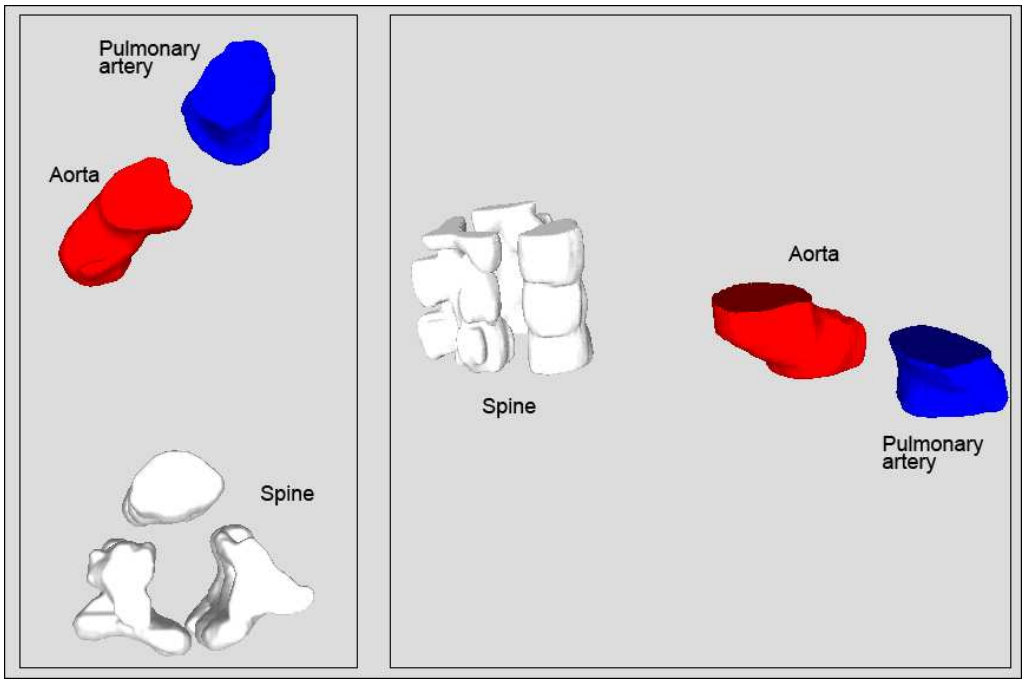


Figure 4.22: Screenshots of 3D reconstruction of Data Set 2.

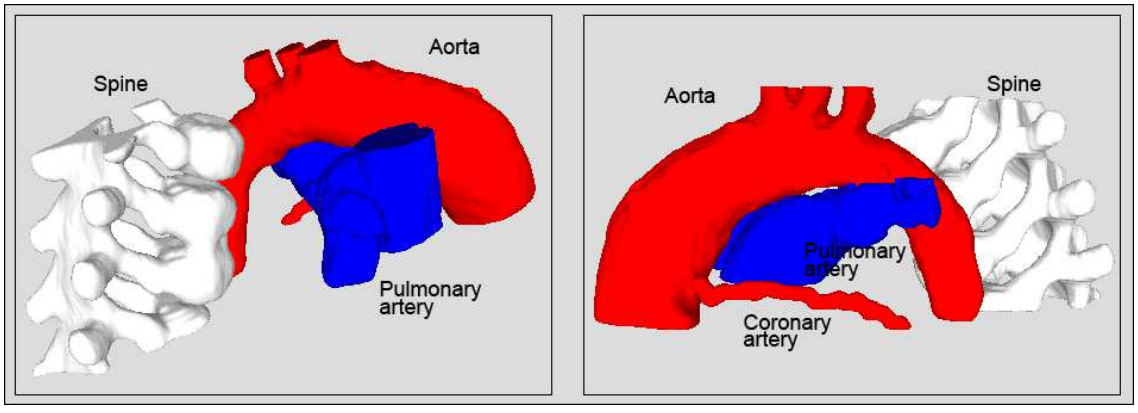


Figure 4.23: Screenshots of 3D reconstruction of Data Set 3.

Recall that in TGA patients, the aorta is located in front of the pulmonary trunk (Section 2.2). Thus, Data Set 1 (Figure 4.21) and Data Set 3 (Figure 4.23) are cases of TGA, whereas Data Set 2 (Figure 4.22) is a normal case. In TGA cases, the positions of ascending aorta and pulmonary trunk are switched.

Additionally, coronary arteries are also visualized except for Data Set 2. As explained in Section 4.4, the coronary arteries in Data Set 2 cannot be seen in the CT images. For Data Set 1 and 3, the positions and orientations of the coronary arteries. However, the outer surfaces are not smooth enough. The possible reasons have already been discussed in Section 4.4.

In 3D visualization, the structures of the cardiac arteries are much clearer than those in 2D images. This will be helpful for surgery simulation. For example, it is much easier to determine the positions to cut the great arteries in the 3D model than 2D images.

Chapter 5

Conclusion and Future Work

In the literature of heart blood vessel segmentation algorithms, most of the algorithms only handle 2D segmentation. Others segment the heart and arteries as a whole in 3D. They do not provide separate segmentation's results of different arteries.

This Thesis describes segmentation and reconstruction algorithms of arteries from CT images for surgical planning simulation system. First, a two-phase level set algorithm segments the great arteries and a geometric parametric model based algorithm is applied to obtain the coronary arteries. Spine is also segmented using region growing for visualization purpose. After that, marching cubes algorithm generates the mesh models from the 3D segmentation results. The analysis of experimental results shows that the algorithms can obtain fairly accurate surface of the great arteries, but the algorithm for coronary artery segmentation seems a bit weak in getting complete and accurate structure of coronary arteries.

The contributions of this thesis are as follows. First, it implemented a set of algorithms for segmenting and reconstructing 3D models of the great arteries and coronary arteries. Second, it discussed the limitations of the segmentation algorithms and how to overcome.

For surgical simulation, it may be useful to segment and build the whole model of the heart that includes all the arteries and veins. Such work is quite challenging, because boundaries between arteries and veins sometimes are unclear or mixed with each other. Level set cannot achieve very satisfactory results without further knowledge, e.g., heart model. On the other hand, more sophisticated algorithms should be employed to segment the coronary arteries. Intensity information, based on which coronary arteries are segmented in this thesis, is not necessarily enough to extract the artery boundaries accurately. Information from other domains, e.g., frequency domain, might also be useful.

So far we have tested algorithms on only three data sets. For further verification of the performance of the algorithms, it would be useful to obtain more data sets from the surgeons to test the algorithms.

Bibliography

[Bal81] D.H. Ballard. Generalizng the hough transform to detect arbitrary shaes. *Pattern Recognition*, 13(2):111–122, 1981.

[BGY+06] C. Bajaj, S. Goswami, Z. Yu, Y. Zhang, Y. Bazilevs, and T. Hughes. Patient specific heart models from high resolution CT. In *Proceedings of the International Symposium on Computational Modelling of Objects Represented in Images*, 2006.

[BP98] A. Bors and I. Pitas. Object segmentation and modeling in volumetric images. In *Proceedings of Workshop on Non-Linear Model Based Image Analysis*, pages 295–300, 1998.

[Cas96] K.R. Castleman. *Digital Image Processing*. Upper Saddle River: Prentice Hall, 1996.

[CLRS01] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press and McGraw-Hill, 2 edition, 2001.

[CPFT98] K. Chandrinos, M. Pilu, R. Fisher, and P. Trahanias. Image processing techniques for the quantification of atherosclerotic changes. In *Proceedings of Mediterranean Conference on Medical and Biological Engineering and Computing*, 1998.

[Dav97] E.R. Davies. *Machine Vision*. San Diego: Academic Press, 1997.

[dBvGB+03] M. de Bruijne, B. van Ginneken, W. Bartels, M.J. van der Laan, J.D. Blankensteijn, W.J. Niessen, and M.A. Viergever. Automated segmentation of abdominal aortic aneurysms in multi-spectral MR images. In *Proceedings of International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 538–545, 2003.

[DLD+05] R. Dindoyal, T. Lambrou, J. Deng, C.F. Ruff, A.D. Linney, C.H. Rodeck, and A. Todd-Pokropek. Level set segmentation of the fetal heart. In *Proceedings of Functional Imaging and Modeling of the Heart*, pages 123–132, 2005.

[EPW+06] O. Ecabert, J. Peters, J. Weese, C. Lorenz, J.von Berg, M.J. Walker, M.E. Olszewski, and M. Vembar. Automatic heart segmentation in CT: current and future applications. *Medicamundi*, 50(3):308–313, 2006.

[ESS+04] S. Eiho, H. Sekiguchi, N. Sugimoto, T. Hanakawa, and S. Urayama. Branch-based region growing method for blood vessel segmentation. In *Proceedings of International Society for Photogrammetry and Remote Sensing Congress*, pages 796–801, 2004.

[FIC04] J. Feng, H. Ip, and S.H. Cheng. A 3D geometric deformable model for tubular structure segmentation. In *Proceedings of Multimedia Modelling Conference*, pages 174–180, 2004.

[FMGW04] C. Florin, R. Moreau-Gobard, and J. Williams. Automatic heart peripheral vessels segmentation based on a normal MIP ray casting technique. In *Proceedings of International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 483–490, 2004.

[GL00] S. Galic and S. Loncaric. Spatio-temporal image segmentation using optical flow and clustering algorithm. In *Proceedings of International Workshop on Image and Signal Processing and Analysis*, pages 63–68, 2000.

[GN94] S.R. Gunn and M.S. Nixon. A model based dual active contour. In *Proceedings of the British Machine Vision Conference*, pages 305–314, 1994.

[GW02] R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. Prentice Hall, 2002.

[HEMK98] K. Haris, S.N. Efstratiadis, N. Maglaveras, and A.K. Katsaggelos. Hybrid image segmentation using watersheds and fast region merging. *IEEE Transactions on Image Processing*, 7(12):1684–1699, 1998.

[HSD73] R.M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, 3(6):610–621, 1973.

[IGR+04] D. Ilea, O. Ghita, K. Robinson, R. Sadleir, M. Lynch, D. Brennan, and P. F. Whelan. Identification of body fat tissues in MRI data. In *Proceedings of International Conference On Optimization of Electrical and Electronic Equipment*, 2004.

[ITK] *Insight Segmentation and Registration Toolkit (ITK). http://www.itk.org.*

[KIF85] J. Kittler, J. Illingworth, and J. Foglein. Threshold based on a simple image statistics. *Computer Vision, Graphics, and Image Processing*, 30:125–147, 1985.

[KQ03] C. Kirbas and F.K.H. Quek. Vessel extraction in medical images by 3D wave propagation and traceback. In *Proceedings of Third IEEE Symposium on Bioinformatics and Bioengineering*, pages 174–181, 2003.

[KRFC05] M. Kalinin, D.S. Raicu, J. D. Furst, and D.S. Channin. A classification approach for anatomical regions segmentation. In *Proceedings of IEEE International Conference on Image Processing*, volume 2, pages 1262–1265, 2005.

[KWT87] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1987.

[LC87] W.E. Lorensen and H.E. Cline. Marching cubes: a high resolution 3D surface reconstruction algorithm. In *Proceedings of International Conference on Computer Graphics and Interactive Techniques, Computer Graphics*, volume 21, pages 163–169, 1987.

[Leo] W.K. Leow. Lecture notes.

[LWK+05] V. Luboz, X. Wu, K. Krissian, C.F. Westin, R. Kikinis, S. Cotin, and S. Dawson. A segmentation and reconstruction technique for 3D vascular structures. In *Proceedings of International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 43–50, 2005.

[Mad97] S.S. Mader. *Inquiry into Life*. McGraw-Hill, 1997.

[MS96] R. Malladi and J.A. Sethian. An O (NlogN) algorithm for shape modeling. In *Proceedings of the National Academy of Sciences*, pages 9389–9392, 1996.

[MST+81] P. Makowski, T.S. Srensen, S.V. Therkildsen, A. Materka, H. Stdkilde-Jrgensen, and E.M. Pedersen. Two-phase active contour method for semiautomatic segmentation of the heart and blood vessels from MRI images for 3D visualization. *Computerized Medical Imaging and Graphics*, 26(1):9–17, 1981.

[NM64] J.A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7:308–313, 1964.

[NYT04] D. Nain, A.J. Yezzi, and G. Turk. Vessel segmentation using a shape driven flow. In *Proceedings of International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 51–59, 2004.

[OS88] S. Osher and J.A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.

[PHS+94] C. Pellot, A. Herment, M. Sigelle, P. Horain, H. Maitre, and P. Peronneau. A 3D reconstruction of vascular structures from two X-ray angiograms using an adapted simulated annealing algorithm. *IEEE Transactions on Medical Imaging*, 13:48–60, 1994.

[PT01] R. Pohle and K.D. Toennies. Segmentation of medical images using adaptive region growing. In *Proceedings of International Conference on Computer Graphics and Interactive Techniques, Computer Graphics*, volume 4322, pages 1337–1346, 2001.

[QK01] F.K.H. Quek and C. Kirbas. Simulated wave propagation and traceback in vascular extraction. In *Proceedings of International Workshop on Medical Imaging and Augmented Reality*, pages 229–234, 2001.

[RCR05] A.B. Redwood, J.J. Camp, and R.A. Robb. Semiautomatic segmentation of the heart from CT images based on intensity and morphological features. In *Proceedings of the International Society for Optical Engineering*, 2005.

[RG01] A. Rao and D. Gillies. Vortex segmentation from cardiac MR 2D velocity images using region growing about vortex centres. In *Medical Imaging and Augmented Reality*, pages 222–225, 2001.

[Set96] J. A. Sethian. *Level Set Methods*. Cambridge University Press, 1996.

[SLC+05] J.V.B. Soares, J.J.G. Leandro, R.M. Cesar, H.F. Jelinek, and M.J. Cree. Retinal vessel segmentation using the 2-D Morlet wavelet and supervised classification. In *Proceedings of Brazilian Symposium on Computer Graphics and Image Processing*, 2005.

[SSV+97] J. Sijbers, P. Scheunders, M. Verhoye, A. van der Linden, D. van Dyck, and E. Raman. Watershed-based segmentation of 3D MR data for volume quantization. *Journal of Magnetic Resonance Imaging : JMRI*, 15(6):679–688, 1997.

[vnl] *Vision Numerics Libraries. http://vxl.sourceforge.net/.*

[WL90] Z. Wu and R. Leahy. A new unsupervised hierarchical segmentation algorithm for textured images. In *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, pages 2325–2328, 1990.

[XP98] C. Xu and J.L. Prince. Snakes, shapes, and gradient vector flow. *IEEE Transactions on Image Processing*, 7:359–369, 1998.

[XP00] C. Xu and J. Prince. Gradient vector flow deformable models. In *Handbook of Medical Imaging*. Academic Press, 2000.