



**Dário Augusto Borges Oliveira**

**A Linear Programming Approach to Vascular Network  
Segmentation from a Single Seed Point**

**TESE DE DOUTORADO**

Thesis presented to the Programa de Pós-Graduação em Engenharia Elétrica of the Departamento de Engenharia Elétrica, PUC-Rio as partial fulfillment of the requirements for the degree of Doutor em Engenharia Elétrica.

Advisor: Prof. Raul Queiroz Feitosa

Rio de Janeiro

April 2013



**Dário Augusto Borges Oliveira**

**A Linear Programming Approach to Vascular Network  
Segmentation from a Single Seed Point**

**TESE DE DOUTORADO**

Thesis presented to the Programa de Pós-Graduação em Engenharia Elétrica of the Departamento de Engenharia Elétrica do Centro Técnico Científico da PUC-Rio, as partial fulfillment of the requirements for the degree of Doutor.

**Prof. Raul Queiroz Feitosa**

Advisor

Departamento de Engenharia Elétrica – PUC-Rio

**Prof. Mauro Monteiro Correia**

INCA

**Profa. Aura Conci**

UFF

**Prof. Aristófanês Corrêa Silva**

UFRJ

**Prof. Bruno Feijo**

Departamento de Informática

**Prof. Sidnei Paciornik**

Departamento de Engenharia de Materiais – PUC-Rio

**Prof. José Eugênio Leal**

Coordinator of the Centro Técnico  
Científico da PUC-Rio

Rio de Janeiro, 12th April 2013.

**Dário Augusto Borges Oliveira**

Holds an Electrical Engineering degree from the Rio de Janeiro State University (UERJ) in 2007, having specialized professionally in the development of medical imaging and remote sensing applications. Obtained the MSc. degree, in Electrical Engineering at the Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) in 2009. Since then has worked in the field of medical imaging and remote sensing image analysis.

Bibliographic data

Oliveira, Dário Augusto Borges

A Linear Programming Approach to Vascular Network Segmentation from a Single Seed Point / Dário Augusto Borges Oliveira; orientador: Raul Queiroz Feitosa. — Rio de Janeiro: PUC-Rio, Departamento de Engenharia Elétrica, 2013.

v., 79 f: il. ; 29,7 cm

1. Tese (doutorado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica.

Inclui referencias bibliográficas.

1. Engenharia Elétrica – Tese. 2. Medical Imaging. 3. Segmentation. 4. Graphs. 5. Linear Programming. 6. Computed Tomography. 7. Vascular Structures Network. I. Feitosa, Raul. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Elétrica. III. Título.

CDD: 510

## Acknowledgments

I would like to thank all who contributed directly or indirectly to the accomplishment of this work: friends, classmates, professors and my family. In special:

To my advisor, Prof. Raul Queiroz Feitosa, for the availability, generosity, encouragement and support throughout the development of my PhD research.

To my advisor during my internship at University of Hannover, Prof. Bodo Rosenhan, for his ideas and discussions. To Laura, Michele and Gerard, for the friendship, laughs and specially for making me feel at home in Hannover.

To Laura Leal Taixé for sharing with me part of the project developed in this thesis. Much of the development was planned during our discussions, and her contributions were fundamental.

To my colleagues from LVC, for the valuable scientific discussions.

To my friends from UERJ, PUC and CPM for showing me how good life is.

To my family, which means everything to me, specially to my parents Dário and Tania, for the unconditional love.

I also gratefully acknowledge the financial support of CNPq and FAPERJ.

## Abstract

Oliveira, Dário Augusto Borges; Feitosa, Raul Queiroz (Advisor). **A Linear Programming Approach to Vascular Network Segmentation from a Single Seed Point**. Rio de Janeiro, 2013. 79p. PhD Thesis — Department of Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

This thesis presents the development and results of this PhD project, which objective, multidisciplinary, was to develop a methodology and a tool for segmenting vascular networks from CT images, using automatic segmentation procedures and visualization of three-dimensional images data. The suggested methodology tracks a vascular network iteratively using a single starting point. The approach uses a conical sampling model composed of multiple concentric and ordered spherical layers. Each sampled point is evaluated using a measurement of vascularity proposed in this thesis, which seeks to identify points that belong to vessels. A directed graph is then built with the selected points and analyzed to find chains of connected points that make up pieces of branches of the vascular network. Each vascular segment found generates a new seed from which a new sampling is performed, and in this way the iterative procedure is repeated until the entire vascular structure is segmented. The methodology was tested using synthetic and real images. Among the real images several different vascular structures were segmented, such as coronary, carotid, hepatic, pulmonary and even a network of nerve fibers in the olfactory system. Vascular network topologies were also identified. The evaluation was quantitative where possible, although this type of data rarely provides a segmentation of reference, and apart from these cases the assessment was qualitative and visual. The results confirm the potential of the method and suggest directions for further developments.

## Keywords

Medical Imaging; Segmentation; Graphs; Linear Programming; Computed Tomography.

## Resumo

Oliveira, Dário Augusto Borges; Feitosa, Raul Queiroz (Orientador). **Segmentação de Redes Vasculares a partir de uma Única Semente Utilizando Programação Linear**. Rio de Janeiro, 2013. 79p. Tese de Doutorado — Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Esta tese apresenta o desenvolvimento e os resultados deste projeto de doutorado, cujo objetivo, de caráter multidisciplinar, foi desenvolver uma metodologia e uma ferramenta para segmentação de redes vasculares a partir de imagens de tomografia computadorizada, utilizando procedimentos de segmentação automática de imagens e visualização tridimensional de dados. A metodologia sugerida segmenta a rede vascular iterativamente utilizando um único ponto de partida. A abordagem utiliza um modelo de amostragem cônico composto de várias camadas esféricas concêntricas ordenadas. Cada ponto amostrado é avaliado utilizando-se uma medida de vascularidade proposta nesta tese, que busca identificar pontos que pertencem a vasos. Um grafo dirigido é então construído com os pontos selecionados e analisado para que se encontre localmente cadeias de pontos conectados que compõem pedaços de ramos da rede vascular. Cada segmento da rede vascular gera uma nova semente a partir da qual uma nova amostragem é realizada e desta forma o procedimento iterativo se repete até que toda a estrutura vascular seja segmentada. A metodologia foi testada utilizando-se imagens sintéticas e reais. Dentre as imagens reais foram segmentadas estruturas vasculares coronárias, carótidas, hepáticas, pulmonares além de uma rede de fibras nervosas do sistema olfativo. Também foram extraídas as topologias das redes vasculares. A avaliação foi quando possível quantitativa, embora este tipo de dado muito raramente ofereça uma segmentação de referência, e nestes casos a avaliação foi qualitativa e visual. Os resultados obtidos confirmam o potencial do método e indicam direções para promover desenvolvimentos futuros.

## Palavras-chave

Imagens Médicas; Segmentação; Grafos; Programação Linear; Tomografia Computadorizada.

# Summary

1	Introduction	<b>12</b>
1.1	Objectives	14
1.2	Thesis contributions	15
1.3	Thesis organization	15
2	Medical Imaging and Vascular Anatomy	<b>17</b>
2.1	Introduction to vascular anatomy	17
2.2	Medical Imagery	18
2.2.1	Computed Tomography	18
2.2.2	DICOM standard	21
3	Theoretical Foundations	<b>23</b>
3.1	Digital Images	23
3.2	Graph Fundamentals	24
3.3	Image Segmentation	26
3.3.1	Data-driven Segmentation	26
3.3.2	Model-driven Segmentation	28
3.4	Network Flow Problem	33
4	Methodology	<b>36</b>
4.1	Motivation and Graph Modeling	36
4.2	Algorithmic Formulation	37
4.3	Sampling Model	39
4.4	Vessel Point Detection	43
4.4.1	Vesselness Computation	43
4.4.2	Vessel Point Candidate Selection	45
4.5	Vascular Network Tracking	46
4.5.1	Vascular Network Detection	47
4.5.2	Tracking with Linear Programming	48
4.5.3	Mapping the linear program to a graphical model	49
4.5.4	Iterative vessel branch detection	51
4.6	Vascular Branches Validation	51
4.6.1	Next seeds definition	52
5	Results and Experimental Analysis	<b>55</b>
5.1	Software Prototypes	55
5.1.1	Batch software	55
5.1.2	GUI software	56
5.2	Segmentation Results Evaluation	58
5.2.1	Synthetic data	59
5.2.2	Pulmonary Data	61
5.2.3	Coronary Data	62
5.2.4	Liver Data	63
5.2.5	Carotids Data	64

5.2.6	Olfactory Projection Fibers (OPF)	65
5.3	Topological Description	66
6	Conclusions	<b>70</b>
6.1	Discussion	70
6.2	Further Research	71
6.3	Final Considerations	72
	References	<b>74</b>



## Figures List

2.1	Vessels anatomy: image depicts the blood flow and the structure of veins and arteries and how they interact. [http://www.baileybio.com/plogger/?level=picture&id=462]	19
2.2	Example of an abdominal CT with the liver shown in green. It is also possible to observe the kidneys, spleen and some other anatomical structures.	20
2.3	Some examples of vessels are highlighted with red circles on the figures, and many other vessels can be observed following the ellipsoidal pattern as well.	21
3.1	CT seen as a tridimensional image.	24
3.2	Example of a graph with seven nodes and weighted edges connecting them.	25
3.3	Dijkstra algorithm step-by-step.	31
3.4	A flow network showing flow/capacity on the edges.	34
3.5	Residual network showing residual capacities.	35
4.1	Modeling a vascular network in a graph using a single direction.	37
4.2	Problems when modeling a vascular network in a graph using a single direction (some edges were hidden to improve understanding).	37
4.3	Graphs built using the local direction of the vascular network	38
4.4	Flowchart of the proposed methodology.	38
4.5	Sampling following a multilayered spheric cone.	40
4.6	Definition of $\alpha$ angle steps to implement $M$ sampling circles at a given layer. It is possible to see the iterative aspect of the process from the steps defined by the sampling distance $s$ .	41
4.7	Vessel point detection steps.	44
4.8	Gaussian mixture for cylinder fitting. The red cylinder fits the vessel at a given $P$ point, and the blue volume models the neighbouring area.	44
4.9	Sample points selection: only the points with high probability of pertaining to a vessel remains in the final graph model.	46
4.10	Network tracking steps.	47
4.11	Finding vascular network branches in a graph with the selected vessel detection points.	47
4.12	Proposed graph structure. First layer of the cone consists of node 1, second layer by nodes 2,3,4 and third layer by nodes 5,6. Each pair of begin/end represents a node referenced by the index associated.	50

4.13	Proposed optimization method example of synthetic images. (a) Initial point marked in orange. First path found is the one with minimum cost (red). In the next step, sources will be added along the path (black arrows). (b,c,d) Paths found iteratively, sources of each path marked by colored arrows. (e) 3D view of the vascular network.	52
4.14	Post processing rules implement some anatomical constraints for a vascular network. The red 'X' shows the branches which would be eliminated following each rule (a) and (b) described in section 4.6.	53
4.15	Next seeds definition.	54
5.1	GUI software - Main screen	57
5.2	GUI Software - Segmentation, Editing and Visualization modules.	58
5.3	Synthetic data segmentation using sinusoidal shaped vascular-like structures	59
5.4	Parameter study of $LP1=D_{radius}$ , $LP2=D_{max}$ , angle in radians and sampling distance. (see sections 4.3 and 4.5)	60
5.5	Segmentation using synthetic data generated by L-Systems	61
5.6	Synthetic data segmentation using L-Systems	62
5.7	Vascular segmentation for pulmonary real dataset	63
5.8	Vascular segmentation for coronary real dataset	64
5.9	This figure shows what we called the blind effect. It is possible to visually understand that since our algorithm does not take into account a specific vessel end point to follow a vessel path, it will not necessarily find a desired vessel, but segment all the vessels connected to the given start point.	65
5.10	Vascular segmentation for liver real dataset	66
5.11	Vascular segmentation for carotids real dataset	67
5.12	Segmentation results for OPF dataset	68
5.13	Extracted topology for synthetic data	68
5.14	Extracted topology for more realistic synthetic data	69

## Tables List

- 5.1 OPF database results. The table shows the results obtained for each exam available in the website. The metric is also the one made available for the competition so as to make it possible a straight comparison with other methods.

67

# 1

## Introduction

The idea of living longer concerns men since ever. Health turned into a main topic for mankind, leading to the development of new healthcare technologies that improved dramatically life expectancy throughout the years. However, since it is not possible to live forever, the challenge posed is to prolong life by preventing the leading causes of death, which are, according to the World Health Organization (WHO), related to vascular diseases and cancer (WHO08).

In Brazil, according to the Ministry of Health, the scenario is similar to the rest of the world. In 2004, from a total of 1.024.073 deaths, 285.543 (27.88%) were related to illnesses in the vascular system, whereas 140.801 (13.75%) were due to tumors (MS04). It is also known that the growth of malignant tumors is directly associated to vessel recruitment and angiogenesis (Holash99). Vascular diseases are related to death and disability in people with diabetes as well (Lüscher03).

The numbers presented above show how urgent it is to develop new technologies to fight and prevent diseases related to vascular issues. Prevention plays a central role, and proper examination tools are indispensable - even when a disease is detected, examination is fundamental to determine the best procedure to be followed.

A great effort to develop this field has been performed, and vascular examination has received special attention. Non-invasive exams, such as Computed Tomography (CT) and Magnetic Resonance Imaging (MRI), are increasingly replacing invasive angiography in examining and visualizing vessels. They are an effective, less traumatic and painless alternative: they allow the detection of vessel anomalies and pathologies such as aneurysms, stenosis and plaques and also provide important information for surgical planning, such as the delineation of vascular supply architecture and operative landmarks.

These kinds of exam, however, provide raw information that must be filtered to deliver useful information. This filtering is usually done visually by a specialist, in a demanding and time consuming task which is also partially subjective. Although the use of contrast chemicals enables specialists to filter the information provided by these exams, surrounding structures, low contrast and artifacts such as stents, usually hinder the correct identification of the structure to be analyzed.

For this reason, segmentation algorithms are welcome to provide support to these specialists.

Many segmentation algorithms for medical imaging have been proposed in the literature. Even so, this is still an open challenge. Depending on the structures the analyst deals with, different approaches are recommended. Deformable models and region growing techniques are used generally to segment organs and massive structures such as bones, muscles and many others (Hegadi10, McInerney96). The combined use of statistical approaches, such as Gaussian mixture models (GMM), and region growing techniques achieved some success in identifying nodules and vessels inside an organ (Oliveira09, Oliveira11).

On the other hand, when it comes to general vessel segmentation, or their topological description, other techniques must be considered (Kirbas04, Lesage09). Many of them are based on a propagating behavior emanating from a given seed point, and they use different techniques that benefit from local information to pursue a vessel path. The differences between them lie on the constraints imposed to the growth process.

The simplest algorithm used is region growing (Wan03, Brien94), which works well for contrasted vessels but leaks into neighbouring structures if contrast is low. To minimize leakage, some algorithms implement the enhancement of vessel-like structures with a pre-processing step using Hessian filters or tubular features (Manniesing04, Canero03). Deformable models, such as level-sets, bring also some robustness against leakage using front propagation to pursue vessels (Lorigo01, Deschamps01), albeit they are not very effective to segment small ones.

Implementing vessel models in a tracking framework is becoming commonplace to deal with vascular structures. The so called vessel tracking algorithms are among the ones with best results for vessel segmentation. They can be divided into ridge, cross-section and tubular tracking. In ridge tracking modeling (Wink00), vessels are modeled as intensity ridges through the use of local information features, such as the Hessian matrix, which allow minimal path approaches to find a minimum cost path between points and identify a vessel (Deschamps01, Olabarriaga03). Differentiation techniques, such the ones using Hessian, are, however, very sensitive to noise, and many times modeling the geometry of a vessel to reinforce vessel constraints is necessary. In cross-section approaches, modeling a vessel cross-section as an ellipse is very usual (Florin05, Krissian06), and provides useful information, such as the local vessel diameter and direction. Again, small vessels are problematic, since their cross-sections may look very different from an ellipse. To overcome this limitation, tubular tracking algorithms (Worz07, Rossignac07) model vessel paths as a chain of connected tubular pieces, since even small vessels keep their tubular aspect. Even though they provide diameter and direction information, and

are more robust for small vessels, they can be computationally very demanding.

Whereas these techniques are reasonably successful for single vessel segmentation, when it comes to vascular network tracking, there is still much room for improvement. The classic minimal path formulation, although widely used, provides only single path minimization. Besides, many of current vessel segmentation algorithms face problems to identify vessel bifurcations, hindering segmentation of vascular networks, and also facing problems to describe their topology. The use of hybrid techniques capable of tracking full networks emerged more recently. For instance, Friman (Friman10) uses a Multiple Hypothesis Tracking framework to connect vessel points described by a vessel template model based on local image and geometry parameters, and Wang (Wang12) proposes a statistical model to track tree-like tubular structures using also a bifurcation detector.

This work proposes a methodology to track vascular structures in non-invasive exams using a graph-based model solved using a linear programming approach.

## 1.1 Objectives

In this thesis a new methodology for tracking and describing the topology of vascular networks is presented. The following objectives were proposed:

**General Objective:** Develop a methodology for tracking and describing the topology of tree-like structures.

**Operational Objectives:**

1. Develop and evaluate a method for tracking vascular structures.
2. Develop tools for visualizing in 2D and 3D the segmented structures.
3. Develop methods for the edition of results.
4. Develop a library containing functions used in this work, to allow the fast prototyping of new related methods.
5. Build a prototype in C++ that implements the techniques developed in this work.

**The following steps for the workflow were proposed:**

1. Construct and select datasets.
2. Develop a pipeline to track iteratively a tree-like structure.
3. Develop a sample model related to vessel morphology.
4. Develop a method based on linear programming to identify vessel paths in a given graph-structured cloud of points.

5. Develop post processing methods to ensure coherence of the segmented structure based on anatomical knowledge.
6. Develop edition tools for correction and completion.
7. Implement the proposed methods in a C++ library.
8. Develop a prototype in C++ and Qt.
9. Evaluate results.

## 1.2

### Thesis contributions

The contributions presented in this thesis are the following:

- Sampling method to provide a cloud of points structured as a graph with geometry that benefits from vessel anatomy.
- Vesselness assessment based on a cylindrical weighted Gaussian mixture model.
- Graph structure modeling to allow the use of a linear programming approach to find vessel paths.
- Iterative pipeline with post processing steps that ensure coherence of the segmented structure and avoid anatomical aberrations.

## 1.3

### Thesis organization

The thesis is structured in six chapters as described bellow:

- Chapter 2 - This chapter introduces the reader to basic concepts of medical imaging and anatomy of vascular systems that will be used throughout the thesis.
- Chapter 3 - The theoretical fundamentals needed for the understanding of this work are shown in this chapter. The main concepts of graph theory, linear programming and network flow problem are discussed.
- Chapter 4 - This chapter the methodology is presented in detail. The algorithmic formulation to track a vascular network iteratively is scrutinized. The sampling model adopted, the vessel points detection methodology and the tracking of vascular structures based on a network flow formulation solved by linear programming are presented in detail.

- Chapter 5 - This chapter depicts the results obtained using images and tables for different datasets, and provides an experimental analysis to discuss efficiency and robustness.
- Chapter 6 - The conclusions are presented in this chapter with a brief discussion over the results obtained and their potential contribution for further research.



## 2

## Medical Imaging and Vascular Anatomy

This chapter presents a brief introduction to vascular anatomy and discusses how vessels appear on the kind of image data used in this work, so as to support the heuristics adopted in the developed methodology.

### 2.1

#### Introduction to vascular anatomy

Blood vessels are part of the cardiovascular system, together with the heart and blood (Silverstein94). They can be simply characterized as the tubes through which blood flows throughout the human body and are involved in almost every medical condition, including cancer, inflammation, and cardiovascular diseases.

Three types of blood vessels are found in the human body: arteries, veins, and capillaries. The functionalities of each are very specific. While arteries typically carry oxygenated blood away from the heart (with the exception of the pulmonary arteries, which carry deoxygenated blood to the lungs), veins carry deoxygenated blood back to the heart, and capillaries are the smallest blood vessels and allow the exchange of various substances such as oxygen, carbon dioxide and nutrients between the bloodstream and the tissues of the body.

Arteries are the largest types of vessels. As they branch away from the heart, they become smaller arterioles, and finally capillaries. The capillaries then connect to venules, then to veins that return blood to the heart and lungs, and then back to the arteries. This way blood flows throughout the body.

Morphologically vessels can vary a lot, but have a similar tubular structure. Arteries are the strongest of blood vessels, carry blood under very high pressure and need to be very strong and elastic. Each artery is comprised of three layers. The outermost layer is the tunica adventia composed by connective tissue and smooth muscle fibers which allow them to handle the force of blood flow. The tunica media is a middle layer composed of muscle cells arranged in a circular pattern to control the constricting and dilating movements of the arteries. The tunica intima is the innermost layer and is composed of three smaller layers that keep the smooth lining of the arteries that allows free blood flow, enclosing the so called lumen, which is the volume occupied by the blood.

Veins also have three layers: tunica externa, tunica media and tunica intima. They do not withstand as much pressure as arteries, so they are much less muscular and elastic, and their walls are much thinner than artery walls. Veins also have valves that avoid blood to flow back in the wrong direction.

Capillaries, as said before, are the smallest of the blood vessels and have walls only one cell thick. These very thin walls allow for selective permeability, allowing oxygen and nutrients that come through the arterioles to cross through capillary walls to the cells of the body. They also allow waste and carbon dioxide to move into the blood that is returning to the heart.

Many disorders can affect blood vessels. The most common are the aneurysm, which is a ballooning caused by a weakness in the vessel wall; the stenosis that occurs when blood clots form in blood vessels causing conditions like a stroke or infarct; and the weakened vessels, which can also disrupt and cause hemorrhage that can be very damaging depending on where it is located. Some of these disorders, such as aneurysm and stenosis, can be detected by imaging since they appear in images usually as abrupt differences in vessels radius, which normally range from few micromilimeters to few centimeters. Image 2.1 depicts the anatomy of vessels.

The next section reports how computed tomography exams (CT) work and how blood vessels appear there. Moreover, it shows the limits imposed by image definition for blood vessel segmentation and how far it is possible to go with current technology.

## 2.2

### Medical Imagery

Medical imaging acquisition techniques provide detailed patient-specific anatomic data, including vascular structures. One of the most established is computed tomography, used in this work. Usually, CT scanners manufactures provide workstations that are used in clinical practice to produce 3D models representing structures identified in the acquired data. However, the segmentation techniques available on these workstations are still too general to provide proper results on vessel segmentation or their topology extraction. To provide methods able to extract this kind of information straight from the acquired image data, it is necessary to understand how human body structures appear in those exams, and more specifically, how vessels appear.

#### 2.2.1

##### Computed Tomography

Computed tomography is a radiological exam that acquires cross-section images of a subject using X-ray measurements taken at different angles around

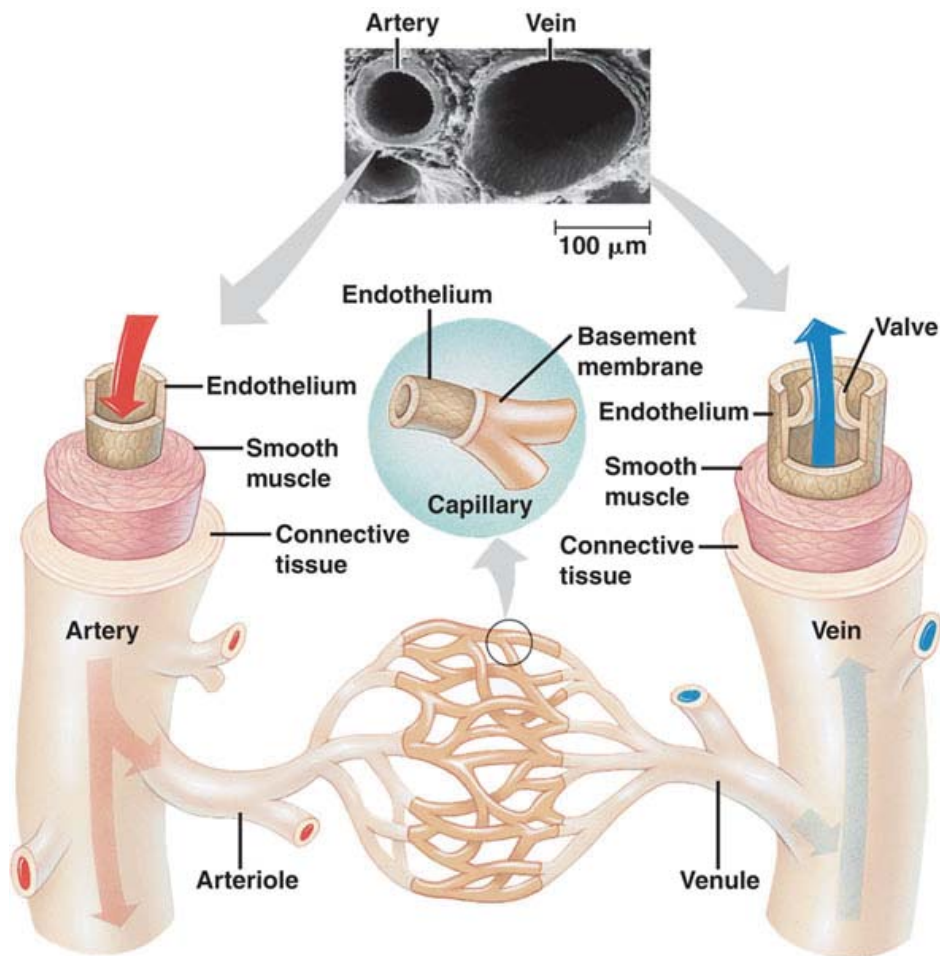


Figure 2.1: Vessels anatomy: image depicts the blood flow and the structure of veins and arteries and how they interact. [http://www.baileybio.com/plogger/?level=picture&id=462]

the patient. Both the radiation source and the detector rotate around the subject to perform a full scan of the patient, who can stay still. The radiation data is obtained through computer-assisted mathematical reconstructions from the data captured by the scanner sensors (Bontrager96). The intensity of X-rays traversed through the body is attenuated according to the density of tissues found in the way, and captured by photon detectors in a way that the line integral of tissue density is measured. This measurement is then represented as a voxel intensity value proportional to the observed attenuation. Image 2.2 shows an example of abdominal tomography. The liver is identified by a green layer.

The tridimensional information is represented as very thin slices of the scanned human body internal structure, with the x-ray beam being collimated strictly to a specific cut at a time, ensuring no overlap images by overlying anatomy. Besides, there's no image degradation by secondary radiation or diffuse tissue outside the cut. Since CT was introduced in clinical practice in the beginning of the 70's, the equipments evolved, and each new scanner generation reduced the

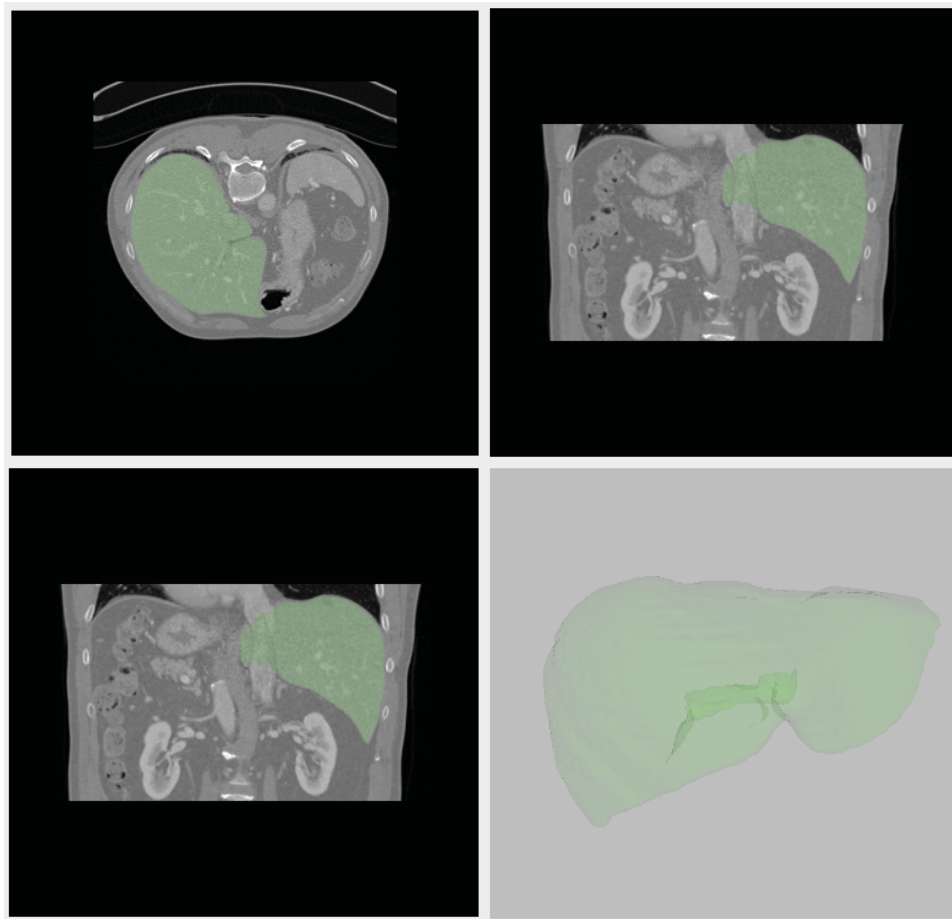


Figure 2.2: Example of an abdominal CT with the liver shown in green. It is also possible to observe the kidneys, spleen and some other anatomical structures.

scan time and the interval between each image reproduced, delivering more accurate exams, with less distortion due to patient's movements or breathing.

In CT images, each voxel (volume element) has a volume size depending on the scanner image definition and slice width. The intensity value at a given position is related to the tissue density there and is defined by a numerical scale in Hounsfield units (HU, named after G. N. Hounsfield, an English scientist responsible for the first cranial TC scan in 1970). After capturing the X-ray beams, a relatively linear attenuation coefficient is determined for each voxel, represented through a matrix of exposure and converted to HU.

In this way a grayscale image is generated, considering as main references the water, assigned with 0 HU (zero); cortical dense bone, that ranges between +1000 and +3000 HU in modern scanners; and the air, that produces the lowest level of attenuation, assigned with -1000 HU (Schneider96). Between the two extremes (air and cortical dense bone) lay different tissues and substances, that have different gray levels according to their density.

Every CT scanner must be calibrated to respect these standard values but it is

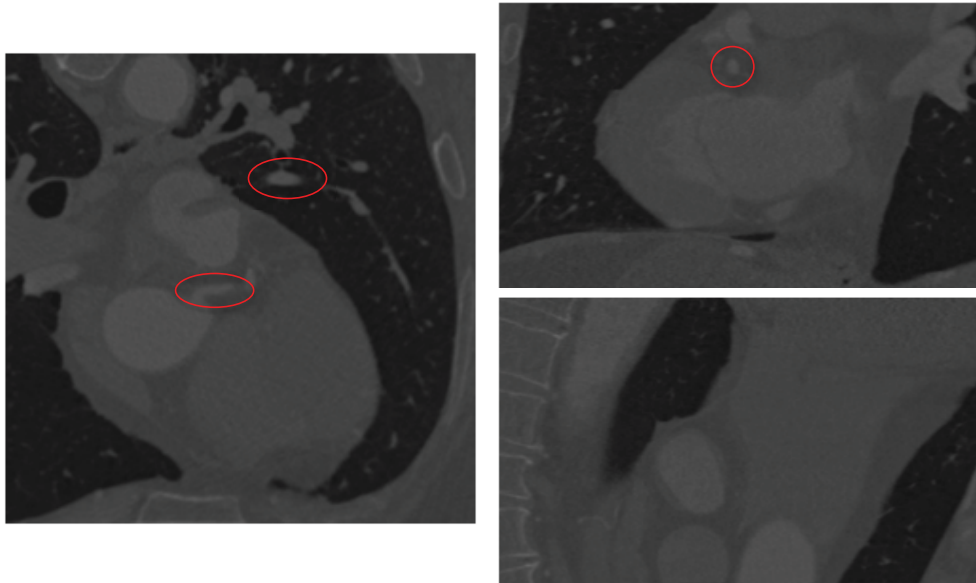


Figure 2.3: Some examples of vessels are highlighted with red circles on the figures, and many other vessels can be observed following the ellipsoidal pattern as well.

also possible to manipulate and adjust the image after the scan, changing brightness, enhancing boundaries, zooming into specific areas and changing contrast to better visualize the desired anatomy. To get an idea, in conventional radiography it is possible to differentiate tissues that have at least 10% of difference in density, whereas in CT exams it is possible to detect differences of 1% or less in tissues density.

Especially for this work it is important to know how vessels appear in CT exams. Blood vessels are filled with blood, which ranges between +30 and +45 HU (Schneider96). This value can vary, however, depending on the calibration of the scanner, and other structures sharing the same range, like muscles (which appear around +40), can hinder their identification. Morphologically they are more or less cylindrical and therefore appear as circles or ellipses depending on the view (axial, coronal or sagittal). Figure 2.3 shows how a vessel usually appears in a CT exam.

### 2.2.2

#### DICOM standard

Different manufacturers produce CT scanners, and as a consequence the exams delivered usually hold small format differences. Hence arose the need to standardize the information format so as to allow the reliable exchange of information between different specialists using CT exams(Bidgood97).

DICOM (Digital Imaging Communications in Medicine) is a program created to implement this standardization. The DICOM standard was built as a group of rules internationally accepted for the communication of biomedical information,

diagnostic and therapeutic. It allows for handling, storing, printing, and transmitting information in medical imaging, and includes a file format definition and a network communications protocol. It is also possible to link patient, hospital and scanner information to the imagery, so as to improve reliability on the provided exam.

This standard was developed by the medical imaging industry, represented by members of NEMA (National Electric Manufacturers Association in USA) and by the community of medical imaging users, such as the American College of Radiology and the European Society of Cardiology. This process has been happening for approximately 15 years, more recent than one could imagine, mainly due to conflicts between the interested parts. The DICOM Standards Committee meets three times a year to propose changes on the standard and adapt it to changing needs.



## 3

### Theoretical Foundations

This chapter describes the theoretical foundations for understanding the methodology proposed in chapter 4 to segment vascular networks. The fundamentals of digital imaging, image segmentation, graphs and network flow analysis are presented.

#### 3.1

##### Digital Images

An image is defined as a bidimensional function,  $f(x, y)$ , where  $x$  and  $y$  are spatial coordinates and the amplitude  $f$  represents the brightness intensity associated to the coordinates  $x$  and  $y$ . Monochromatic images follow exactly this model, whereas coloured images are composed by a combination of 2D images. For instance, in RGB format, there are three 2D components that represents each one a primary color (red, blue and green) that are combined to form a complex color at each pixel.

Images can be analogical, i.e., be continuous with respect to the  $x$  and  $y$  axis, and to the amplitude  $f$  as well. However, in computer vision the usual is to work with digital images, which are represented by a matrix containing real numbers. With this kind of images it is possible to perform the diverse mathematical operations that allow the extraction of useful information from imagery.

CT exams, used in this work, deliver 3D digital imagery that can be viewed as a regular stack of 2D images. In this way, each exam can be represented as a function  $f(x, y, z)$ , where  $x$ ,  $y$  and  $z$  are the spatial coordinates and  $f$  is the intensity assigned to  $(x, y, z)$ . Moreover, the voxel is usually parallelepiped-shaped since the scale of  $x$  and  $y$  differs from the scale in  $z$ . The stack of 2D images composes a volume, as represented in figure 3.1, which preserves spatial relationship with the real scanned body, allowing the estimation of organ volumes, density measurements and many other morphometric and radiometric measurements from the human body.

#### 3.2

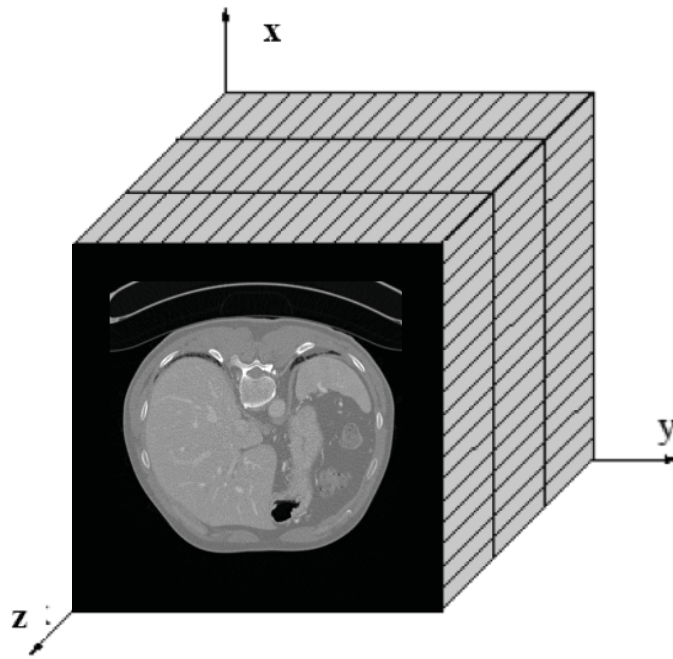


Figure 3.1: CT seen as a tridimensional image.

## Graph Fundamentals

Graphs are mathematical structures used to model pairwise relationships between objects from a given collection (Diestel10). In image processing for instance, these objects can be pixels, segments or features of certain regions. A graph is composed by a collection of vertices or nodes and a collection of edges that connect pairs of vertices. The nodes may be part of the graph structure, or may be external entities represented by integer indices or references. A graph data structure may also associate to each edge some edge value, such as a symbolic label or a numeric attribute. Figure 3.2 shows an example of graph.

A graph is commonly represented as an ordered pair  $G = (V, E)$ , comprising a set  $V$  of vertices or nodes together with a set  $E$  of edges or lines. Edges are two-element subsets  $E(x, y)$  of  $V$ , represented also by the somewhat shorter notation  $xy$ . The order  $|V(G)|$  of a graph is the number of its vertices, and the size  $|E(G)|$  of a graph is the number of its edges. The graph  $G$  may be undirected, meaning that there is no distinction between the edges  $xy$  and  $yx$ , or its edges may be directed from one vertex to another, meaning that  $xy$  and  $yx$  represent different edges. The two endpoints of an edge are also said to be adjacent to each other.

A loop in graph theory is an edge whose endpoints are the same vertex, whereas a link has two distinct endpoints. An edge is multiple if there is another edge with the same endpoints, otherwise it is simple. The multiplicity of an edge is the number of multiple edges sharing the same end vertices; the multiplicity of a graph, the maximum multiplicity of its edges. A graph is a simple graph if it has no



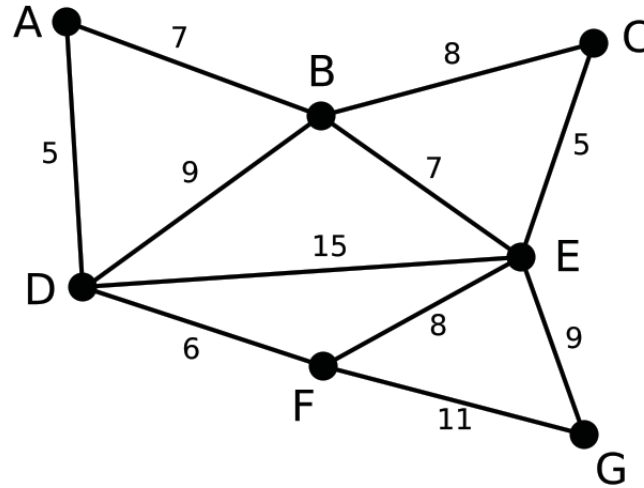


Figure 3.2: Example of a graph with seven nodes and weighted edges connecting them.

multiple edges or loops.

A non-edge is an edge that is not present in the graph. More formally, for two vertices  $u$  and  $v$ ,  $\{u, v\}$  is a non-edge in a graph  $G$  whenever  $\{u, v\}$  is not an edge in  $G$ . The complement of a graph  $G$  is a graph with the same vertex set as  $G$  but with an edge set such that  $E(x, y)$  is an edge in the complement only if  $xy$  is not an edge in  $G$ . An edgeless graph or empty graph or null graph is a graph with zero or more vertices, but no edges.

A walk is an alternating sequence of vertices and edges, beginning and ending with a vertex, where each vertex is incident to both the edge that precedes it and the edge that follows it in the sequence, and where the vertices that precede and follow an edge are the end vertices of that edge. A walk is closed if its first and last vertices are the same, and open if they are different. A path is an open walk, and usually understood to be simple, meaning that no vertices are repeated. The closed equivalent to this type of walk, a walk that starts and ends at the same vertex but has no repeated vertices or edges, is called a cycle.

A weighted graph associates weight (also called cost), usually real numbers, to every edge in the graph. The weight of a path or the weight of a tree in a weighted graph is the sum of the weights of the selected edges. A weighted graph can be also called a network.

Graphs structured as networks are specially suited for vascular segmentation, and will be used in this thesis, since it possible to model topological knowledge about vessels in this kind of structure, as we describe further.

### 3.3

## Image Segmentation

Human vision considers certain observable characteristics of objects in a scene, such as shape and color to form groups within a context established by the image, decomposing the image into significant elements in a process called segmentation.

Image segmentation is a process that seek to emulate human cognitive processes to identify relevant objects in the image, and is usually the most delicate step in image processing. It is still subject of intense research, highly dependent on the problem posed, and therefore usually based on prior knowledge about the application field.

In medical imaging, medicine provides the knowledge about human body anatomy used to develop heuristics that guides segmentation methods combining various image processing techniques. These techniques can be basically divided into two different groups (Masutani06): data-driven and model-driven.

### 3.3.1

#### Data-driven Segmentation

Data-driven segmentation techniques (Fujimoto02, Kim00) try to emulate the human ability to identify visually objects, by detecting and classifying segments and features in images using information over similarity present in the data. They can be grouped into contour based segmentation and region based segmentation.

#### Contour Segmentation

This segmentation model relies on the fact that different regions often form groups with different levels of intensity, which ultimately generate on their frontiers a sharp variation of intensity, i.e., a contour. The analysis of contours is usually very interesting for segmenting objects. For instance, generally, boundaries of objects in a scene present variation of intensity, otherwise these objects would not be visible in the image. Repeated patterns of edges can also be used on the definition of heuristics for segmentation, indicating, for example, different textures for different objects.

If an object has edges detectable in its entire contour, then contour based approaches are usually enough for having good segmentation results. It is very usual to take advantage of borders detectors, such as the Canny detector (Canny86), to improve reliability and achieve good segmentation results using contour.

However, in many cases, such as vascular structures in CT exams, it is not possible to clearly define an edge along the entire length of the object to be segmented. Neighbour anatomical structures sharing more or less the same density of vessels, such as muscles (see chapter 2), compromise the clear definition of a

complete contour and make it difficult to achieve good results using only traditional contour based segmentation techniques.

## Region Segmentation

Objects of interest in an image are usually composed by segments or regions. Formally, segments can be defined as sets of interconnected pixels that share some common characteristics such as similar texture or intensity. Region segmentation methods aims to identify objects in an image by decomposing it into similar regions according to some criterion of homogeneity.

There is a large number of segmentation methods based in regions, such as Baatz&Schappe (Vanessa10) and Spring (Camara96), each of them is suited for a group of problems. For vessel segmentation many techniques are suited, and one of the most basic is region growing.

## Region Growing by Hysteresis

If one is interested in segmenting vessels without considering their topology, then region growing is quite effective. This method segment objects by fusing pixels or subregions into larger regions iteratively. The procedure can be described as the following:

1. Select an image pixel or region belonging to the object to be segmented. It is called a seed.
2. Merge neighboring pixels or subregions to the segmented region if they satisfy some criterion of homogeneity such as similar intensity, color or texture.
3. Repeat step 2 until no more neighbouring pixel or subregion meet the similarity criterion.

This process is based on a greedy strategy and can be slow depending on image size and the homogeneity criterion used. An option for speeding up it is to group pixels previously in regions by computing intervals of intensity that characterize somehow a particular concept or structure in the image. This approach is known as region growing by hysteresis, and follows a very simple procedure to define seeds and the group of pixels that can be merged to them.

In this formulation, seeds are defined by a strict range of gray levels, narrowly defining pixels that surely belong to a particular concept. Then, a second range, more comprehensive, determines a larger group containing pixels that will be added to the concept only if they are both similar to the seeds and belong to their neighborhood.

The procedure is then defined by five steps:

1. Label seeds as groups of adjacent pixels whose intensities belong to the range of intensity  $[I_L, I_H]$ . These pixels will necessarily belong to the final segmented object.
2. Label a set of candidates for the object to be segmented, with less certainty. They are defined as the pixels belonging to the range of intensity  $[I_L - \Delta_L, I_H + \Delta_H]$ . Pixels with intensity value outside this range will not belong to the final segmented object.
3. Define a initial solution composed by the seeds.
4. Add the candidates adjacent to the current solution in 3 dimensions.
5. Repeat the previous step until the object estimated stops growing, ie, until the resulting volume remains the same from one iteration to the next.

The resulting object is formed by pixels whose intensity is certainly characteristic for the object to be segmented (seeds), and by pixels that both lie in the seeds vicinity and have intensity values acceptable for describing the object to be segmented. This model is particularly attractive for regions with intensity dispersion or when the intensity value of pixels belonging to the concept varies according to their position in space, like blood vessels in CT images.

### 3.3.2

#### Model-driven Segmentation

Well contrasted structures are usually identified without a great effort using data-driven methods. However, when it comes to images with low contrast, i.e., when adjacent structures share pixel intensity values, they usually fail. More information about the object model represented in the image, such as the shape, is needed and model-driven approaches allow the embedding of this information in the segmentation process.

Model-driven segmentation methods (Lamecker05, Soler01, Friman10) use pre-defined templates to segment objects in images by matching the model to the data. In this type of technique, the model representing the structure to be segmented is defined by means of object characteristics such as spatial position, texture and spatial relationship with other objects. Once these features are defined, the algorithm searches the image for instances that fit the given model.

The different model-driven methods proposed rely on the kind of model used, which can be probabilistic, geometrical or deformable. Among them, for vessel

segmentation, the use of shortest path approaches (Deschamps01, Olabarriaga03) is almost commonplace.

The shortest path problem is formally defined as the one of finding the optimal path between two points, for instance pixels in an image, according to some objective criteria. The optimization is usually defined as the cost minimization of penalties associated to the edges that compose the path.

Transposing this to graph theory, let  $e_{i,j}$  be the edge incident to both vertices  $v_i$  and  $v_j$ . Given a real-valued weight function  $f : E \rightarrow \mathbb{R}$ , and an undirected simple graph  $G$ , the shortest path from  $v$  to  $v'$  is the path  $P = (v_1, v_2, \dots, v_n)$  (where  $v_1 = v$  and  $v_n = v'$ ) that over all possible  $n$  minimizes the sum  $\sum_{i=1}^{n-1} f(e_{i,i+1})$ . When the graph is unweighted this is equivalent to finding the path with fewest edges.

Considering the various formulations for addressing the shortest path problem, two can be stated: the fast marching method, an special case of level-sets formulation proposed by Sethian and Osher (Sethian99, Osher03); and the graph-based Dijkstra's algorithm (Ahuja93). The second one is specially interesting for this thesis approach, as it is going to be clear in the following sections.

### Dijkstra's algorithm

In 1959 the Dutch computer scientist Edsger Dijkstra published a graph search algorithm that solves the single-source shortest path for a graph with non-negative edge path costs. The solution is composed by the shortest paths from a source vertex  $v$  to all other vertices in the graph.

The algorithm is composed by conceptually simple steps (Ahuja93). Suppose one wants to find the shortest path between two nodes in a graph. To start, the distance to every node in the graph is marked with infinity or unvisited, which does not imply there is an infinite distance, but only that the nodes were not visited yet. Then, at each iteration, a current node is selected, beginning with the starting point, for which the distance is marked as zero. For subsequent iterations, the current node is set as the closest unvisited node to the starting point, which is easily computed by inspection of the visited nodes.

At each iteration, an update of the distance to every unvisited node directly connected to the current node is performed. This is done by computing the sum of the distance between the unvisited node and the distance value of the current node. If the sum value is less then the current value of the unvisited node then it replaces the current value. In effect, the unvisited node is relabeled if the path to it through the current node is shorter than the previously known paths. Nodes marked as visited are labeled with the shortest path from the starting point to it and will not be revisited or returned to.

The iterative process continues updating the neighboring nodes with the shortest distances, marking the current node as visited, and moving onto the closest unvisited node until the destination node is visited. In some sense, this algorithm implements a search outward from the starting point, iteratively considering every node that is closer in terms of shortest path distance until it reaches the destination. It clearly finds the shortest path, however it may be relatively slow in some topologies.

If the process is run until every node is visited then the paths with lowest cost (i.e. the shortest path) between a source node and every other node are found. If one is interested in finding only the shortest path between two nodes, then the iterative process can be stopped once the destination node is marked as visited, which should be faster.

Algorithmically it states like this:

1. Assign to every node a temporary distance value, initial node is set to zero and all other nodes to infinity.
2. Mark all nodes as unvisited, and set the initial node as current. Create a set of the unvisited nodes called unvisited set consisting of all the nodes except the initial node.
3. For the current node, consider all of its unvisited neighbors and calculate their temporary distances. For example, if the current node  $y$  is marked with a distance of 5, and the edge connecting it with a neighbor  $z$  has length 2, then the distance to  $z$  (through  $y$ ) will be  $5+2=7$ . If this distance is less than the previously recorded temporary distance of  $z$ , then overwrite that distance. Even though a neighbor has been examined, it is not marked as "visited" at this time, and it remains in the unvisited set.
4. When all unvisited neighbors of the current node were visited, mark the current node as visited and remove it from the unvisited set. A visited node will never be checked again.
5. If the destination node has been marked visited (when planning a route between two specific nodes) or if every node is already marked as visited (when planning a complete traversal), then stop. The algorithm has finished.
6. Select the unvisited node that is marked with the smallest temporary distance, and set it as the new "current node" then go back to step 3.

In figure 3.3 it is possible to follow the algorithm stepwisely for the definition of the shortest paths to all other nodes and their respective final costs, i.e., final distance values. Starting from node  $s$ . The initial costs are set to zero for node  $s$  and



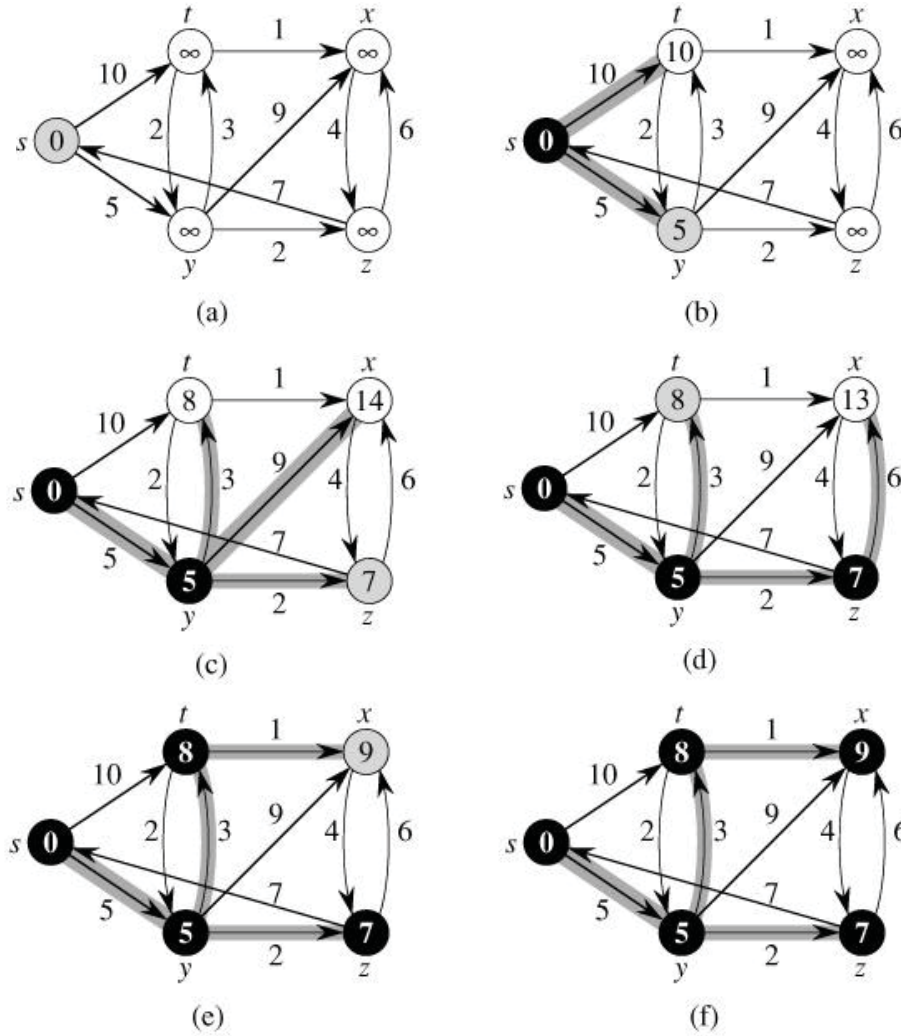


Figure 3.3: Dijkstra algorithm step-by-step.

infinity for all the others. Then the adjacent nodes  $t$  and  $y$  have their cost values updated in step (b) considering the edges costs linking them and  $s$ , but not flagged as visited yet. In next step the unvisited node with lowest cost (node  $y$ ) is selected, set as visited, and its adjacent nodes are checked to update their cost values. One can notice that in step (c) node  $t$  has its value updated again, since the path through  $y$  has a lower cost than the path coming straight from  $s$ . The procedure is repeated in steps (d), (e) and (f), until all nodes are flagged as visited. The gray shadow shows the shortest paths found from node  $s$  to all the others.

Concerning computation efficiency, Dijkstra's original algorithm does not use a min-priority queue and runs in  $O(|V|^2)$ , but there is an implementation based on a min-priority queue implemented by a Fibonacci heap and running in  $O(|E| + |V| \log |V|)$ . This is asymptotically the fastest known single-source shortest-path algorithm for arbitrary directed graphs with unbounded non-negative weights.

#### – Linear programming formulation

Linear programming (LP) is a mathematical methodology for optimizing a mathematical model represented by linear relationships (Dantzig63). More formally, LP is a technique for the optimization of a linear objective function, subject to linear equalities and linear inequalities. The linear inequalities implement constraints to the problem and define half spaces that together compose a convex polyhedron representing the feasible geometric place for the problem. The linear objective function is a real-valued affine function defined on this polyhedron, and the linear programming algorithm finds a point in this region where the function has the smallest (or largest) value (if such a point exists).

A general LP problem can be expressed in canonical form:

$$\begin{cases} \text{maximize} & \mathbf{c}^T \mathbf{x} \\ \text{subject to} & A\mathbf{x} \leq \mathbf{b} \\ \text{and} & \mathbf{x} \geq \mathbf{0} \end{cases} \quad (3-1)$$

where  $\mathbf{x}$  represents the vector of variables (to be determined),  $\mathbf{c}$  and  $\mathbf{b}$  are vectors of (known) coefficients,  $A$  is a (known) matrix of coefficients, and  $(\cdot)^T$  is the matrix transpose. The expression to be maximized or minimized is called the objective function ( $\mathbf{c}^T \mathbf{x}$  in this case). The inequalities  $A\mathbf{x} \leq \mathbf{b}$  are the constraints which specify a convex polyhedron over which the objective function is to be optimized.

The shortest path problem can be naturally associated to a linear programming formulation. Formally, given a directed graph  $G(V, E)$  with source node  $s$ , target node  $t$ , and cost  $w_{ij}$  for each edge  $(i, j)$  in  $E$ , the linear program with variables  $x_{ij}$  represents the shortest path problem:

$$\begin{cases} \text{minimize} & \sum_{ij \in E} w_{ij} x_{ij} \\ \text{subject to} & x \geq 0 \\ \text{and for all } i, & \sum_j x_{ij} - \sum_j x_{ji} = \begin{cases} 1, & \text{if } i = s; \\ -1, & \text{if } i = t; \\ 0, & \text{otherwise.} \end{cases} \end{cases} \quad (3-2)$$

Many methods were proposed in the literature for solving this kind of linear equations system. A very common approach is the simplex algorithm. It was proposed by George Dantzig in 1947 (Dantzig63) and proceeds by performing successive pivot operations on the system of linear equations system which each give an improved basic feasible solution. The choice of pivot element at each step



is determined so as that this pivot improves the solution. The algorithm is quite efficient and can be guaranteed to find the global optimum if certain precautions against cycling are taken. However, the simplex algorithm has poor worst-case behavior for some problems, where it takes a number of steps exponential to the problem size and can become unmanageable.

From a geometrical point of view, simplex solves LP problems by finding a feasible solution at a vertex of the inequalities based polyhedron and then walking along a path on the edges of the polyhedron to other vertices until an optimum is reached. This walk happens with non-decreasing values of the objective function, and in many practical problems, "stalling" occurs and many steps are made with no increase in the objective function. The reader can refer to (Dantzig63) for a detailed explanation of the method.

The shortest path formulation is very used for single vessel segmentation. However, for vascular network segmentation, a more complex model, based for instance in network flow theory, is more appropriate, since we are dealing with many connected paths forming a network.

### 3.4 Network Flow Problem

A flow network is a directed weighted graph where each edge receives a flow, albeit the amount of flow on it cannot exceed its capacity (Ahuja93). Besides, a flow must satisfy the restriction that the amount of flow into a node equals the amount of flow out of it, except when it is a source, which has more outgoing flow, or a sink, which has more incoming flow. This kind of structure can be used to model traffic in a road system, fluids in pipes, or anything similar in which something travels through a network of nodes. We model a vascular network this way.

Let  $G(V, E)$  be a directed graph in which every edge  $(u, v) \in E$  has a non-negative, real-valued capacity  $c(u, v)$ . If  $(u, v) \notin E$ , it is assumed that  $c(u, v) = 0$ . There are two special nodes: a source  $s$  from which the flow emanates and a sink  $t$  in which the flow goes out of the network. A flow network is the real function  $f : V \times V \rightarrow \mathbb{R}$  with the following three properties for all nodes  $u$  and  $v$ :

1. Capacity constraints:  $f(u, v) \leq c(u, v)$ . The flow along an edge cannot exceed its capacity.
2. Skew symmetry:  $f(u, v) = -f(v, u)$ . The net flow from  $u$  to  $v$  must be the opposite of the net flow from  $v$  to  $u$ .
3. Flow conservation:  $\sum_{w \in V} f(u, w) = 0$ , unless  $u = s$  or  $u = t$ . The network flow to a node is zero, except for the source, which produces flow, and the sink, which consumes flow.

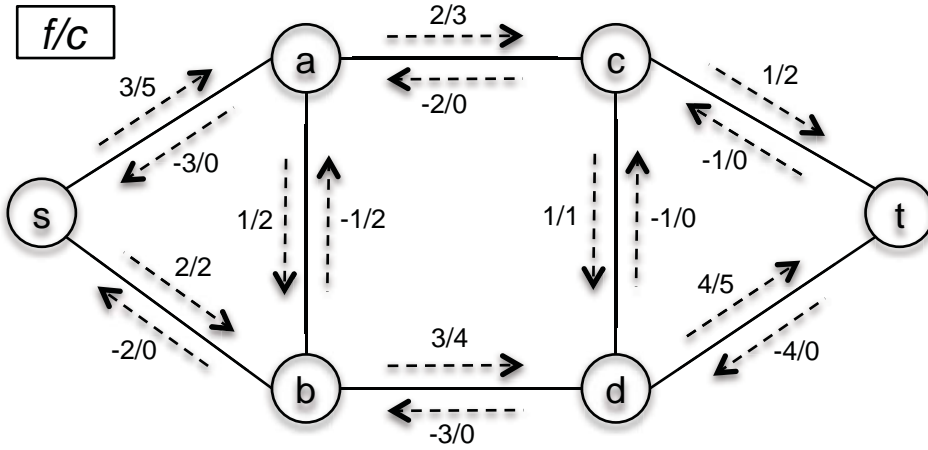


Figure 3.4: A flow network showing flow/capacity on the edges.

Formally,  $f(u, v)$  is the net flow from  $u$  to  $v$ . For example, if there is a flow of 4 units from  $u$  to  $v$ , and a flow of 3 units from  $v$  to  $u$ , one says that  $f(u, v) = 1$  and  $f(v, u) = -1$ .

Image 3.4, shows an example of a flow network with source  $s$ , sink  $t$ , and four additional nodes. The flow and capacity are denoted  $f/c$  and are represented over the edges. Notice that the total amount of flow from  $s$  to  $t$  is 5, since the total outgoing flow from  $s$  is 5, which is also the incoming flow to  $t$ .

The residual capacity of an edge is  $c_f(u, v) = c(u, v) - f(u, v)$ , and it defines a residual network denoted  $G_f(V, E_f)$ , which gives the amount of available capacity of a network. Since flows in opposite directions cancel out, decreasing the flow from  $v$  to  $u$  is the same as increasing the flow from  $u$  to  $v$ . An augmenting path is a path  $(u_1, u_2, \dots, u_k)$  in the residual network, where  $u_1 = s$ ,  $u_k = t$ , and  $c_f(u_i, u_{i+1}) > 0$ . A network is at maximum flow if and only if there is no augmenting path in the residual network.

Figure 3.5 shows the residual network for the given network flow example. Notice how there is positive residual capacity on some edges where the original capacity is zero, for example for the edge  $(d, c)$ . This flow is not a maximum flow.

The network flow architecture holds the modeling of many different problems and solutions depending on the applications. The most common problem using flow networks is to find the maximum flow, which provides the largest possible total flow from the source to the sink in a given graph, and can be solved efficiently with the Ford–Fulkerson algorithm (Ahuja93). The max-flow min-cut theorem (Ahuja93) states that finding a maximal network flow is equivalent to finding a cut of minimum capacity that separates the source and the sink, and can be solved in a linear programming formulation as well.

Another common configuration is the minimum cost flow, used in this thesis to track a vascular network. Here, each edge  $u, v$  has a given cost  $c(u, v)$ , and the

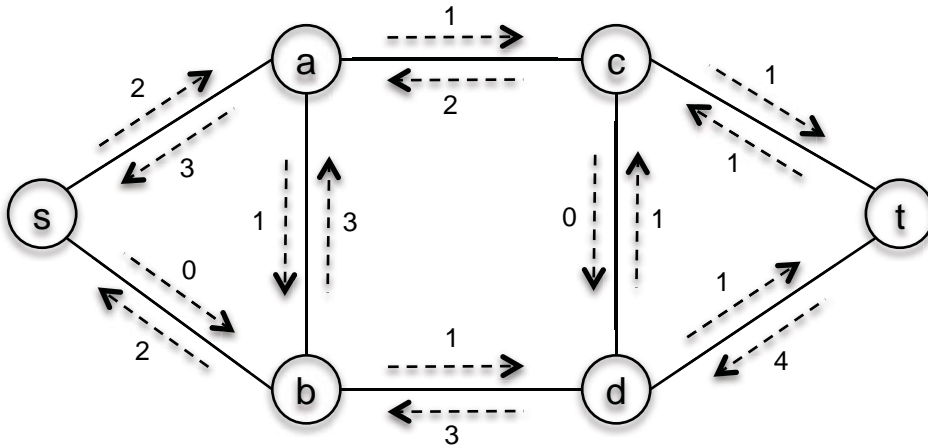


Figure 3.5: Residual network showing residual capacities.

cost of sending the flow  $f(u, v)$  through the edge is  $f(u, v) \cdot c(u, v)$ . The objective then is to send a given amount of flow from the source to the sink, at the lowest possible cost. It is possible to see this problem as a generalization of the shortest path problem algorithmically solved by Dijkstra, which could be modeled with the starting node being the source  $s$ , the destination node the sink  $t$  and the shortest path the one with the lowest total cost between these two nodes.

Next chapter details our methodology for segmenting a whole vascular network using a single seed point. We propose an iterative method that allows multiple vessel segmentation using a minimum cost network flow formulation solved by linear programming.

## 4 Methodology

This chapter covers the proposed methodology for tracking a vascular network from a given starting point. The first section (4.1) presents some motivation for the proposed methodology and shows how vessels can be modelled as directed graphs. Section 4.2 shows the general processing steps for the iterative segmentation of vascular networks, whereas section 4.3 addresses the sampling method based on anatomical characteristics of vessels. Detection of vessel points is described in section 4.4, while section 4.5 describes the local vascular network tracking procedure, using a network flow model solved by linear programming. Finally, section 4.6 presents validation steps that ensure the anatomical coherence of the segmented vascular network.

### 4.1 Motivation and Graph Modeling

The use of directed graphs to segment tree-like structures such as vascular networks is quite intuitive. Figure 4.1 shows how a directed graph composed by vessel point detections can describe a vascular structure. They allow the segmentation of branches of a vascular network through the identification of linked vessel point detections, such as the path depicted in green.

Not very intuitive is the problem that arises from this choice when it comes to tracking vascular structures in images whose voxels are distributed uniformly in a given direction, such as CT scans. Vessels usually change direction continuously and hardly follow the direction of an axis  $x$ ,  $y$  or  $z$ . Since directed graphs, as shown in section 3.2, consist of nodes organized in interconnected layers following a certain direction, this behaviour is problematic. The use of any of  $x$ ,  $y$ ,  $z$  axes to define the directed graph levels, would lead to the impossibility of segmenting vessels whose direction goes in favor of a given axis at some point, but against it later. In other words, the directed graph modeling does not allow a path passing through a sequence of levels  $N$  and  $N+1$  to return to any node on level  $N$ . Figure 4.2 depicts the problem. The yellow branch cannot be totally identified, due to a change of direction that disrespects the directed graph structure. The detection stops at the third vessel point detection of that branch, since the path is not allowed to go back

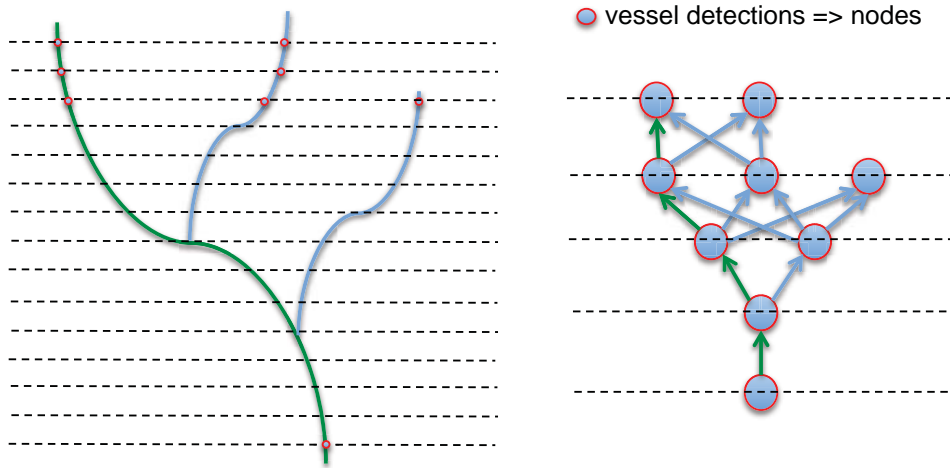


Figure 4.1: Modeling a vascular network in a graph using a single direction.

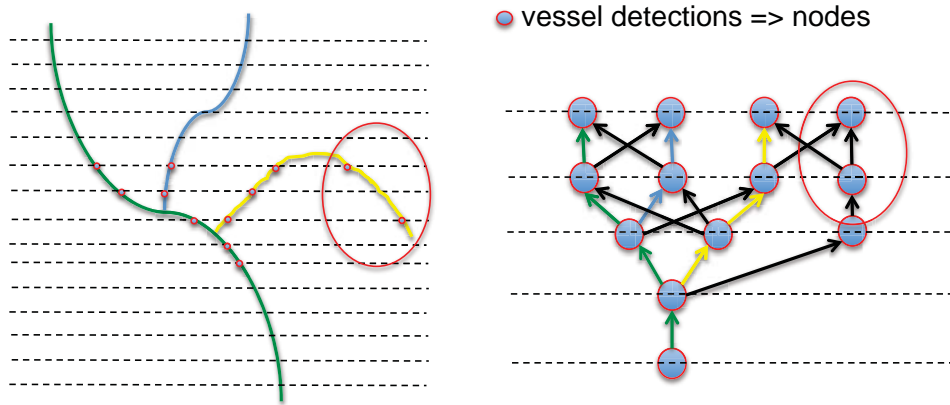


Figure 4.2: Problems when modeling a vascular network in a graph using a single direction (some edges were hidden to improve understanding).

on graph levels.

This thesis proposes an alternative solution for this problem, through the definition of several local graphs, modeled according to the vessel direction at a given point. The model for the graph implements local coordinates whose rotation and translation vary according to the position and the direction of the vascular network at a given point, as shown (simplified) in figure 4.3. This model provides, at least locally, an environment in which the problem described above rarely occurs and thus where the use of directed graphs is suitable.

The procedure that implements this special sampling follows a conical model, which derives a valid directed graph, and is formally defined in section 4.3.

## 4.2 Algorithmic Formulation

The methodology proposed for finding full vascular networks is implemented iteratively through a sequence of steps that uses a single starting point and provides

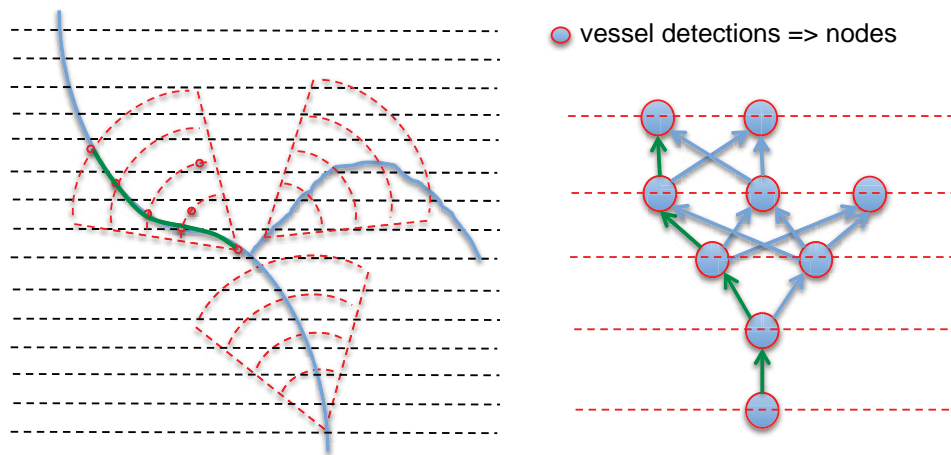


Figure 4.3: Graphs built using the local direction of the vascular network

the tracking of a vascular network and its topology as well. It is composed by four steps, as shown in the flowchart depicted in figure 4.4.

The iterative process can be understood with the aid of figures 4.3 and 4.4. Using a user-defined starting point, part of the vascular network is identified and defines new seeds, which feed the iterative process until the full vascular network is segmented, that means, until no new seed is found. It is interesting to notice that this proposition is a natural consequence of the use of local directed graphs for modeling the vascular structure, but it is at the same time, what allows their use for tracking vascular networks.

The formal definition of the algorithm states as the following:

1. Definition of the input exam, parameters and initial seed.
2. Sampling from a given seed point.
3. Detection of vessel points among sample points.
4. Tracking of vascular network within the detected vessel points.
5. Definition of new seeds from the vascular structure detected.
6. If there is any seed to evaluate, go back to step 2.

Each of these steps is explained in detail in the following sections.

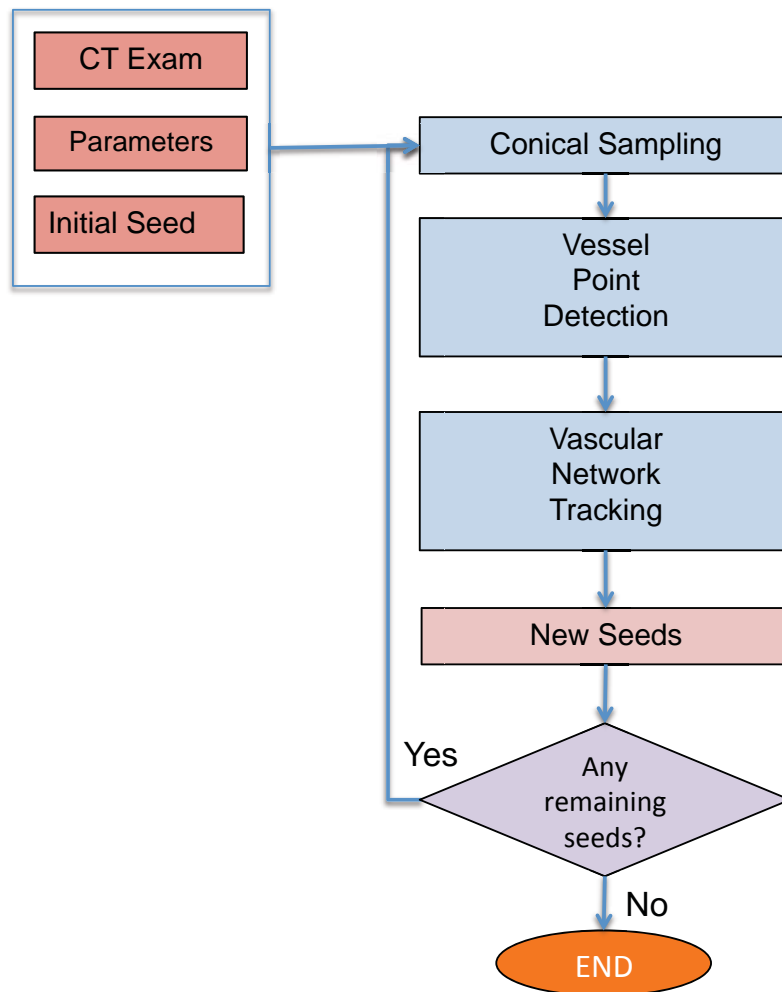


Figure 4.4: Flowchart of the proposed methodology.

### 4.3 Sampling Model

The sampling method follows a conical model composed by multiple concentric and ordinated spherical layers, as figure 4.5 illustrates. This model has some interesting characteristics for sampling vessel structures, such as:

- Points belonging to the same layer are equidistant in relation to the origin. This provides an equiprobability (with respect to the coordinates) of points laying on the same layer to belong to the same vessel found at the origin.
- The surface area of layers increases with the distance to the origin. It is taken into account that the increase of distance in relation to the origin, increases also the potential of spatial spreading of a vessel in relation to the estimated direction axis.
- The cone opening angle and height are parameters to be set. Their definition is related to the anatomic likelihood of a specific kind of vessel to change direction suddenly.

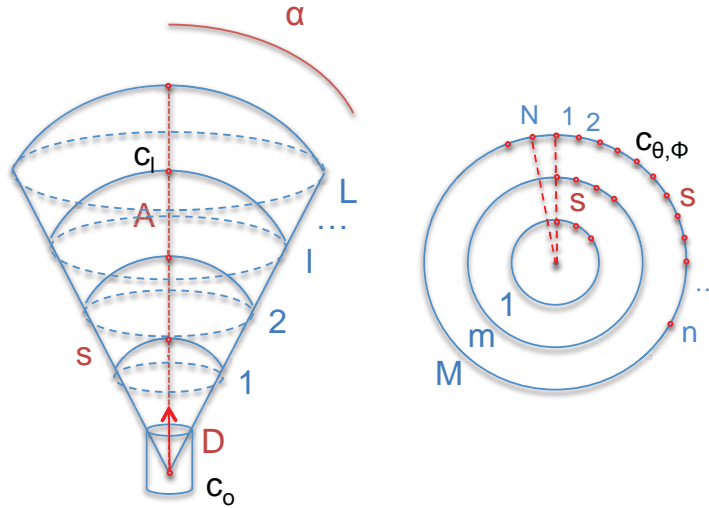


Figure 4.5: Sampling following a multilayered spheric cone.

- The sampling distance is isometric. Its definition can be related to the vessel radius size and the voxel size.

The geometric model proposed for sampling points is defined by four parameters: the unitary direction vector  $D$ , estimated by a cylinder fit on the given seed point (see section 4.4); the aperture angle  $\alpha$ , the axis length  $A$ , and the sampling distance  $s$ , defined empirically and set by the user. This four parameters define a cone with spherical basis that delimits the volume to be sampled.

Figures 4.5 and 4.6 depict the model in detail. The sample points are grouped in classes depending on how they are computed:  $c_o$  is the origin,  $c_l$  are points on the center line  $A$  and  $c_{\theta, \phi}$  are points on the concentric circles of each layer.

The number of layers  $L$  (equation 4-1), the number of concentric circles  $M$  (equation 4-3) in each layer, and the number of points  $N$  (equation 4-8) on each circle from a given layer are computed respecting the user-defined sampling distance  $s$ , so as to deliver isometric sample points. It is possible to notice that these equations are just fractionating the space uniformly.

$$L = \frac{A}{s} \quad (4-1)$$

$$A_l = \frac{A * l}{L} \quad (4-2)$$

$$M = \frac{\alpha * A_l}{s} \quad (4-3)$$



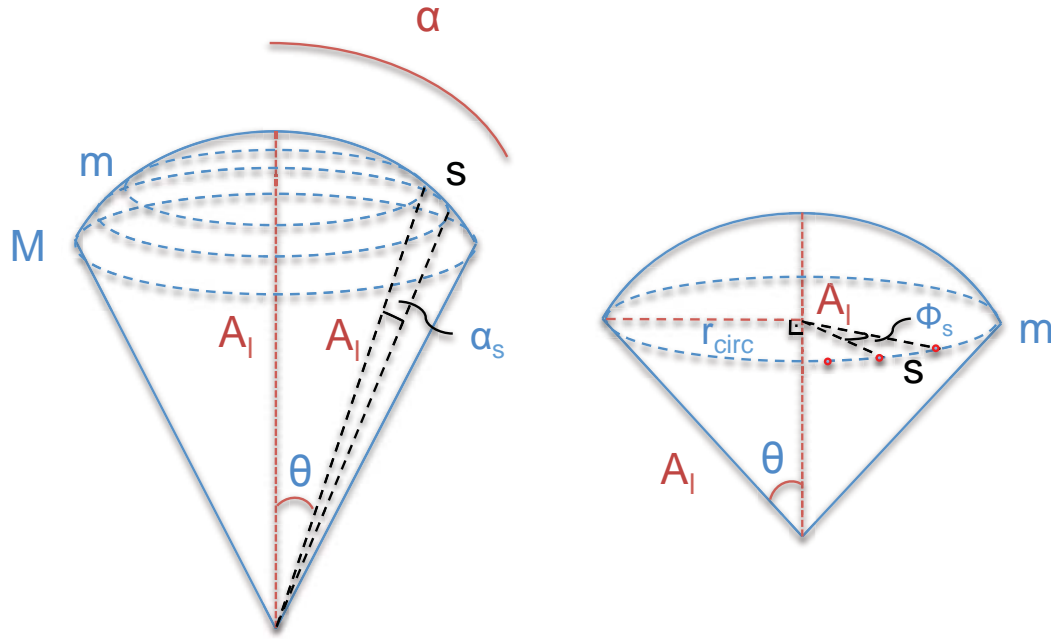


Figure 4.6: Definition of  $\alpha$  angle steps to implement  $M$  sampling circles at a given layer. It is possible to see the iterative aspect of the process from the steps defined by the sampling distance  $s$ .

$$c_l = c_o + A_l \cdot D \quad (4-4)$$

The axis size  $A_l$  corresponding to layer  $l$  is defined in equation 4-2. The number  $M$  of circles at a layer  $l$  is computed respecting the sampling distance  $s$  as well: each circle is defined by a corresponding  $\alpha_s$  step using the length of the arc  $\alpha * A_l$  delineated by the angle  $\alpha$  at layer  $l$ . The left model on figure 4.6 illustrates the idea. This spatial division fractionates the spherical calotte composing a layer  $l$  in evenly distributed circles where  $c_{\theta,\phi}$  sample points lay on.

$$\theta = \frac{\alpha * m}{M} \quad (4-5)$$

Each circle  $m$  is defined by a corresponding  $\theta$  angle given by equation 4-5, and the number of points  $N$  laying on the circle is defined simply using trigonometry rules, as shown in the right model of figure 4.6. The circle  $m$  corresponding circumference length is computed and partionated using the angular step  $\phi_s$  to deliver points with sampling distance  $s$ . The variables involved are defined below.

$$\phi_s = \frac{s}{r_{circ}} \quad (4-6)$$

$$r_{circ} = A_l \cdot \sin \theta \quad (4-7)$$

$$N = \frac{2\pi \cdot r_{circ}}{s} \quad (4-8)$$

Once we define a cloud of points following the proposed geometry, it is necessary to rotate and translate their coordinates according to the cone direction and origin, so as to place them at the right spatial position, as described in section 4.1. This is performed by a regular transformation matrix.

For rotation, it is important to observe some considerations. The computed cloud of points is fairly symmetric with respect to the Z-axis, and therefore, any new coordinate basis where the original Z-axis is aligned with the direction  $D$  is suitable for our model. In such case an orthonormal transformation matrix  $H$  has the third row equal to  $D$ , which is an unitary vector.

This first restriction defines an infinite number of bases with one of the axes equal to  $D$ , and since any of them is suitable, we can determine an arbitrary second axis considering only the restriction to be orthogonal to  $D$ , i.e.,  $D \cdot V_1 = 0$ . The third axis for the new basis is defined straightforwardly as the vectorial product between the two other axes.

Formally, let the direction axis be  $D = [D_x D_y D_z]$ . An arbitrary unitary axis  $V_1$  orthogonal to  $D$  is defined in equation 4-9 and the rotation matrix is filled with a third axis  $V_2 = D \times V_1$ , as shown in equation 4-11.

$$[V_1] = \begin{bmatrix} -D_y & D_x & 0 \end{bmatrix} \quad (4-9)$$

$$[V_2] = \begin{bmatrix} -D_z D_x & -D_z D_y & D_x D_x + D_y D_y \end{bmatrix} \quad (4-10)$$

$$[H] = \begin{bmatrix} \frac{-D_y}{|V_1|} & \frac{D_x}{|V_1|} & 0 \\ \frac{-D_z D_x}{|V_2|} & \frac{-D_z D_y}{|V_2|} & \frac{D_x D_x + D_y D_y}{|V_2|} \\ D_x & D_y & D_z \end{bmatrix} \quad (4-11)$$

The cloud of  $c_{\theta, \phi}$  points are computed using the angles  $\theta$  and  $\phi$  from the geometric model (refer to figures 4.5 and 4.6), computed for each concentric circle

**Algorithm 1** Iterative sample point coordinates definition

---

```

Add point  $c_o$  to  $B$  (Seed point at cone origin).
Compute  $L$  using equation 4-1
for  $l = 1$  to  $l = L$  do
    Compute  $A_l$  using equation 4-2.
    Compute  $c_l$  using equation 4-4.
    Add  $c_l$  to  $B$ . (Points laying on the axis  $A$ ).
    Compute number  $M$  of circles of layer  $l$  using equation 4-3.
    for  $m = 1$  to  $m = M$  do
        Compute angle  $\theta$  and radius  $r_{circ}$  corresponding to the circle  $m$  using formulas 4-5
        and 4-7 respectively.
        Compute  $\phi_s$  using equation 4-6.
        for  $\phi = 0$  to  $\phi = 2 \cdot \pi$  do
            Compute  $c_{\theta,\phi}$  using equation 4-12.
            Add  $c_{\theta,\phi}$  to  $B$ . (Points laying on the  $M$  circles of a layer  $l$ ).
             $\phi = \phi + \phi_s$ 
        end for
    end for
end for

```

---

at a given layer, and then rotated and translated using matrix  $H$  and the origin  $c_o$ , as defined in equation 4-12.

$$\begin{bmatrix} c_{\theta,\phi} \end{bmatrix} = H \begin{bmatrix} A_l \begin{bmatrix} \cos \phi \cdot \sin \theta \\ \sin \phi \cdot \sin \theta \\ \cos \theta \end{bmatrix} \end{bmatrix} + c_o \quad (4-12)$$

where  $H$  is the rotation matrix that brings the sample points to direction  $D$ ,  $\phi$  is defined by equation 4-6,  $\theta$  is defined by equation 4-5,  $A_l$  by equation 4-2 and  $c_o$  is the seed point at the origin of the cone.

The computation of final coordinates for every sampled points follows an iterative procedure. Let  $B$  be the set of sample points coordinates. Algorithm 1 is composed of a few steps that fill  $B$  sequentially.

Once we defined all sample point coordinates, each of them is evaluated as a vessel point candidate, using a vesselness measurement proposed in section 4.4. The selected ones are structured as a directed graph using the conical layers structure as described in section 4.5.

## 4.4

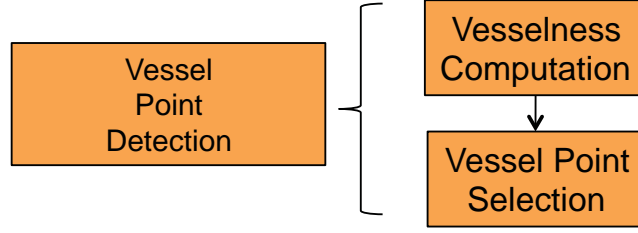


Figure 4.7: Vessel point detection steps.

## Vessel Point Detection

This section shows how sample points are selected as points that actually pertain to vessel path. This is done in two steps, as depicted in figure 4.7. First, we compute a measurement, hereby called vesselness, to assess how likely it is for a sample point to be part of a vessel. Then, we select only points that have local characteristics, including vesselness, that are peculiar to vessels. These steps are shown in detail in the following.

### 4.4.1

#### Vesselness Computation

As stated in section 4.1, the nodes in our graph are associated to vessel hypotheses, and therefore a model that delivers a vesselness measurement at each node must be defined. In the literature, a number of different models have been suggested, e.g., elliptical cross-sections models (Florin05), spheres (Rossignac07) or template models (Friman10, Worz07). Here, we propose a model for vesselness computation using two concentric cylinders based on the common assumption that a vessel and its neighboring volume form locally a biphasic Gaussian mixture. Intuitively, this measurement seems to match the description of a vessel in CT images: fairly cylindrical and with different intensity values from the surrounding structures.

The cylindrical model is defined by a center point  $P$ , a direction of the cylinder and a inner radius  $R_2$  (whose value is meant to be the vessel radius). The central point is given by the position of the sample point, while the cylinder direction and radius are the parameters to be defined during the cylinder fit process.

Let  $Q_1$  and  $Q_2$  be two concentric cylinders with radius,  $R_1$  and  $R_2$ , as shown in figure 4.8.  $W_{Q_{in}}$  represents the intensity of voxels  $t$  lying inside cylinder  $Q_1$ , and  $W_{Q_2}$  the voxels lying inside cylinder  $Q_2$ . Let  $F_{in}$  represent the Gaussian distribution estimated from  $W_{Q_{in}}$  and  $F_{out}$  represent the Gaussian distribution estimated from  $W_{Q_{out}} = W_{Q_2} \setminus W_{Q_{in}}$ . The vesselness is then given by:

$$W_m = \frac{\sum_{t \in W_{Q_{in}}} F_{out}(t) + \sum_{t \in W_{Q_{out}}} F_{in}(t)}{\|W_{Q_2}\|} \quad (4-13)$$

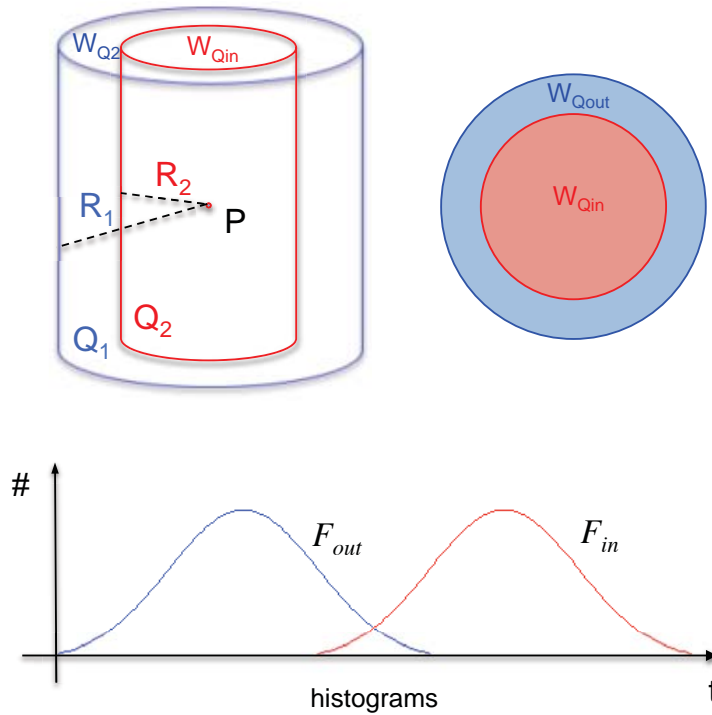


Figure 4.8: Gaussian mixture for cylinder fitting. The red cylinder fits the vessel at a given  $P$  point, and the blue volume models the neighbouring area.

The given vesselness formula represents how much the inner and outer volume voxel intensities are not within other Gaussian distribution, i.e., how exclusive each distribution is in relation to the other. In this sense, a good cylinder fit would define the inner and outer cylinders for which  $F_{in}$  and  $F_{out}$  form a biphasic mixture, and therefore deliver very low values as an outcome. Since vessels are more or less cylindrical structures filled with blood and surrounded by tissues, usually non-filled with blood, it is reasonable to assume this as a vesselness measurement.

The cylinder direction and radius must be defined to model the cylinder that fits the best to the vessel at a given point. For this, a stochastic method called differential evolution (Das11) was used. The outcome delivers the vesselness measurement and the optimal cylinder radius and direction for the given point. It is important to notice that any other stochastic method could be used, and that the optimization technique chosen is not really the focus in this thesis.

#### 4.4.2

##### Vessel Point Candidate Selection

Now that we have a cloud of sample points  $B$  and a measurement for evaluating them as vessel points, it is necessary to define some rules to select those with great chance of actually belonging to a vessel. The selected points are then stored, along with their vesselness assessment, in a graph structure as shown (simplified) in figure 4.9, respecting the models presented in sections 4.3 and 4.5.

The points selection is composed of two steps:

1. Points whose intensity values differs much from the ones observed at the origin of a given sampling cone are discarded and not even evaluated for cylinder fit optimization. This avoids waste of processing time, since these points are not likely to hold vessels. It is implemented by estimating the intensity Gaussian distribution of both the neighborhood of the seed point and of the point to be evaluated, and checking how similar they are.
2. Points selected in step 1 are submitted to a cylinder fit. The cylinder found by the optimization process is then validated to ensure it follows some rules, defined below.

The second step uses three basic rules:

1. The mean value of the inner cylinder must be higher than the mean value of the outer volume (see figure 4.8 for details). We assume that vessels appear as bright cylindrical structures in CT images.
2. The vesselness for the evaluated point must be higher than a fraction of the vesselness at the cone origin. This is done to avoid very badly evaluated points to go further and hinder or slower the network tracking process. This proportion is a user defined parameter and works somehow as a sensitivity factor, which allows for finding weaker vessels (at the cost of adding noise) or just stronger vascular branches.
3. Vessel points whose cylinder radius is too small are discarded. It happens that the optimization process tends to deliver cylinders with very small radius when the point is actually not in a vessel. Therefore, these points are eliminated, and the threshold for eliminating them is the image spacing, which is quite reasonable, since vessels with radius lower than image spacing would hardly be detectable.

Once vessel point detections are defined, a graph is built and analyzed to identify paths forming a vascular network, as explained in next section.

## 4.5 Vascular Network Tracking

The formulation for tracking vascular networks structured in graphs is presented in this section as a minimum-cost network flow problem, efficiently solved using linear programming. It is composed of two steps as depicted in figure 4.10: vascular network detection and vascular network validation.

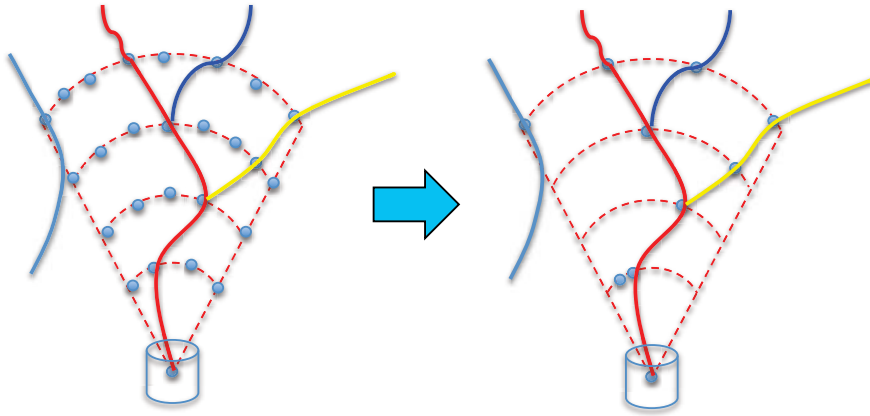


Figure 4.9: Sample points selection: only the points with high probability of pertaining to a vessel remains in the final graph model.

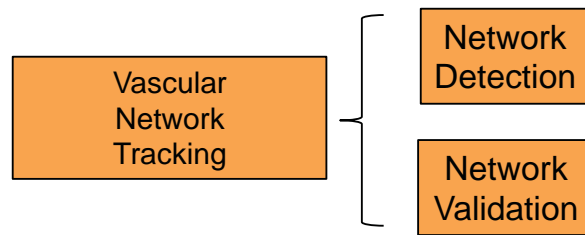


Figure 4.10: Network tracking steps.

#### 4.5.1 Vascular Network Detection

The idea for detecting vascular networks is to build and analyze local graphs for which the nodes represent the vessel detections, as found in Section 4.4, using the computed sample points coordinates and their respective vesselness value. These nodes are fully connected to neighboring detections by edges, which determine the relation between two detections with an assigned cost. Thereby, the matching problem is equivalent to a minimum-cost network flow problem: finding the optimal set of vessels is equivalent to sending flow through the graph so as to minimize the cost.

Let  $\mathcal{O} = \{\mathbf{o}_i\}$  be a set of vessel detections with  $\mathbf{o}_i = (x_i, y_i, z_i)$  represented by the 3D position. A vessel is defined as a list of ordered vessel detections  $V_k = \{\mathbf{o}_{k_1}, \mathbf{o}_{k_2}, \dots, \mathbf{o}_{k_N}\}$ , where  $k_1, k_2, \dots, k_N$  follow the axis direction found in Section 4.4. A set of vessels  $\mathcal{V}$  composing the vascular network is defined to be the optimal set of vessels  $\mathcal{V}^* = \{V_k\}$  that best explains the detections. This is equivalent to maximizing the a-posteriori probability of  $\mathcal{V}$  given the set of detections  $\mathcal{O}$ . Assuming detections are conditionally independent, the objective function is expressed as:

$$\mathcal{V}^* = \underset{\mathcal{V}}{\operatorname{argmax}} P(\mathcal{V}|\mathcal{O}) = \underset{\mathcal{V}}{\operatorname{argmax}} \prod_i P_{\text{prob}}(\mathbf{o}_i|\mathcal{V})P(\mathcal{V}), \quad (4-14)$$

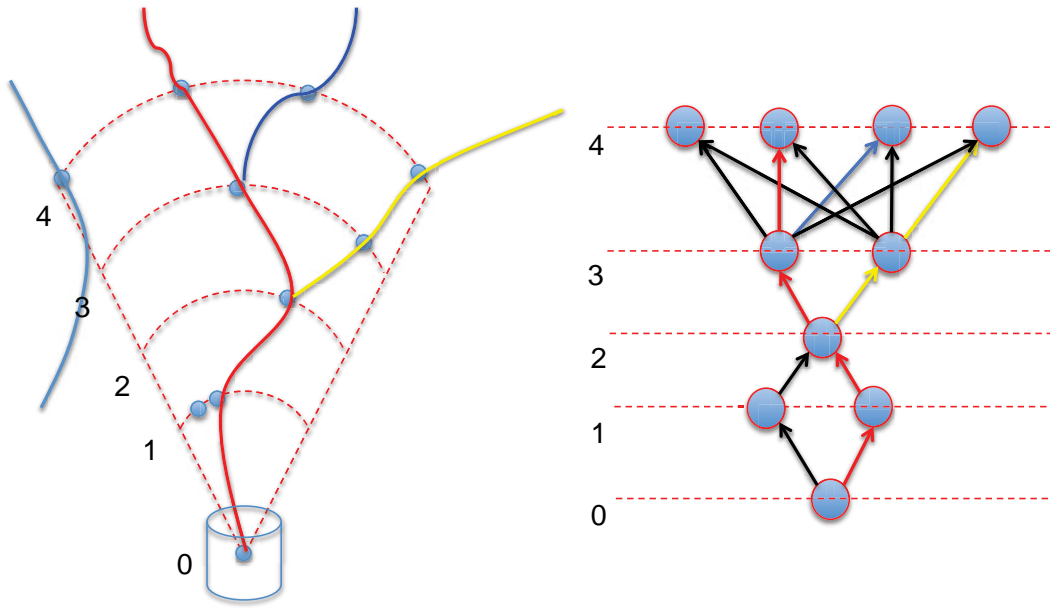


Figure 4.11: Finding vascular network branches in a graph with the selected vessel detection points.

where  $P_{\text{prob}}(\mathbf{o}_i|\mathcal{V})$  is the likelihood of the detection (vesselness measure) as explained in Eq. (4-13). In order to reduce the space of  $\mathcal{V}$ , we make the assumption that vessels cannot overlap (i.e., a detection cannot belong to two vessels). The only case in which this will not hold is at a vessel bifurcation. We will deal with this special case with a new proposed optimization scheme presented in subsection 4.5.4. We also define each vessel to be independent, i.e., each is found independently of the others, which leads to the decomposition:

$$P(\mathcal{V}) = \prod_{V_k \in \mathcal{V}} P(V_k) = \prod_{V_k \in \mathcal{V}} P_{\text{in}}(\mathbf{o}_{k_1}) \dots P_{\text{t}}(\mathbf{o}_{k_i}|\mathbf{o}_{k_{i-1}}) \dots P_{\text{out}}(\mathbf{o}_{k_N}) \quad (4-15)$$

for each vessel  $k$ , represented by an ordered chain.  $P_{\text{in}}(\mathbf{o}_i)$  or  $P_{\text{out}}(\mathbf{o}_i)$  is the probability that a trajectory starts or ends with detection  $\mathbf{o}_i$ .  $P_{\text{t}}(\mathbf{o}_i|\mathbf{o}_j)$  is the probability that  $\mathbf{o}_j$  is followed by  $\mathbf{o}_i$  in the trajectory. As shown in next subsection, these probabilities are going to be described as products of costs  $C$  and flows  $f$ , with no detriment to the optimal path to be found during the minimization process.

#### 4.5.2

##### Tracking with Linear Programming

We linearize the objective function by defining a set of flow flags  $f(i)$  which indicate if an edge  $i$  is in the path of a trajectory or not. In a minimum cost network flow problem, the objective is to find the values of the variables that minimize the total cost of the flows over the network. Defining the costs as negative log-likelihoods, and combining Equations (4-14) and (4-15), the following objective



function is obtained:

$$\begin{aligned} \mathcal{V}^* &= \underset{\mathcal{V}}{\operatorname{argmin}} -\log P(V) + \sum_i -\log P_{\text{prob}}(\mathbf{o}_i|\mathcal{V}) \\ &= \underset{f}{\operatorname{argmin}} \sum_i C_{\text{in}}(i)f_{\text{in}}(i) + C_{\text{out}}(i)f_{\text{out}}(i) + C_{\text{prob}}(i)f_{\text{prob}}(i) + \sum_{i,j} C_t(i,j)f_t(i,j) \end{aligned} \quad (4-16)$$

subject to the following constraints:

- Edge capacities: we assume that each detection belongs either to one vessel or to none, i.e.  $0 \leq f(i) \leq 1$  in its linearly relaxed form.
- Flow conservation at the nodes: the entering flow of a node equals its exiting flow.

$$f_{\text{in}}(i) + f_{\text{prob}}(i) = \sum_j f_t(i,j) \quad \sum_j f_t(j,i) = f_{\text{out}}(i) + f_{\text{prob}}(i) \quad (4-17)$$

This way, we have a fully defined linear program. In the next section, we represent this formulation as an intuitive graphical model, and explain each of the terms present in equation 4-17.

### 4.5.3

#### Mapping the linear program to a graphical model

To map this formulation into a cost-flow network, we define  $G = (N, E)$  to be a directed network with a cost  $C(i)$  associated with every edge  $i \in E$ . An example of such a network is shown in Fig. 4.12; it contains two special nodes, the source  $S$  and the sink  $T$ ; all flow that goes through the graph starts at the  $S$  node and ends at the  $T$  node; each flow represents a vessel  $V_k$ . Each observation  $\mathbf{o}_i$  is represented with two nodes, the beginning node  $b_i \in N$  and the end node  $e_i \in N$  (see Fig. 4.12) and a detection edge connecting both nodes.

It is important to highlight that each vessel detection point is represented in the graph by a pair of nodes  $b_i$  and  $e_i$  connected by an edge with a cost  $C_{\text{prob}}(i)$  associated to the vesselness measurement for the sampled point. Also, each vessel detection point is connected to the next layer detection points  $(e_i, b_j)$  by edges with a cost  $C_t(i, j)$  associated to the coordinates distance between the two points in question. In this way an optimal path would minimize the costs associated to both vesselness and distance between nodes composing the path.

Below we detail four types of edges present in the graphical model. The edges associated to  $C_{\text{toll}}(i, j)$  are going to be explained later.

**Transition edges.** The edges  $(i, j)$  connect the end nodes  $e_i$  with the beginning nodes  $b_j$  in the following layers (orange edges in Fig. 4.12), with cost  $C_t(i, j)$

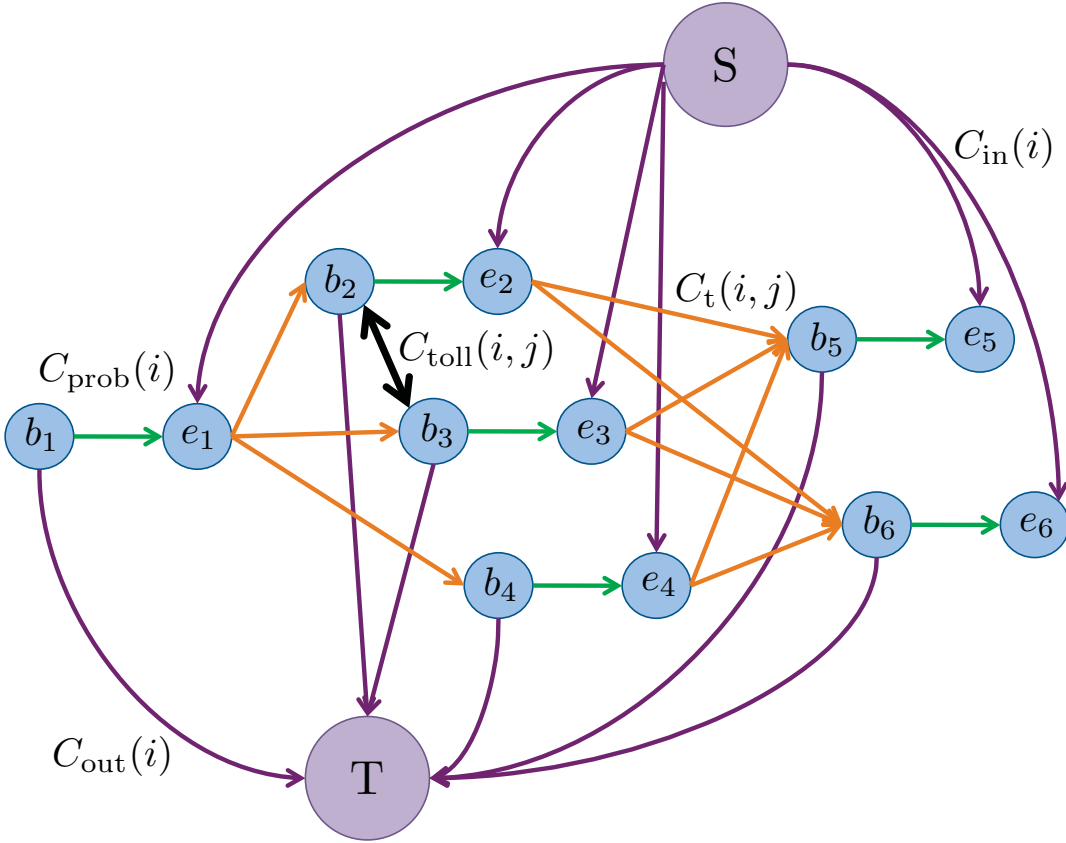


Figure 4.12: Proposed graph structure. First layer of the cone consists of node 1, second layer by nodes 2,3,4 and third layer by nodes 5,6. Each pair of begin/end represents a node referenced by the index associated.

and flow  $f_t(i, j) = 1$  if  $\mathbf{o}_i$  and  $\mathbf{o}_j$  belong to  $V_k$ , and 0 otherwise. The costs of the transition edges represent the spatial relation between different points in the vessel. Since we are interested in matching points close to each other, we define the costs to be a linear decreasing function of the distance between neighboring vessel detections, assuming a maximum allowed distance  $D_{\max}$ :

$$C_t(i, j) = -\log(P(\mathbf{o}_j|\mathbf{o}_i)) = -\log\left(\frac{D_{\max} - \|\mathbf{o}_j - \mathbf{o}_i\|}{D_{\max}}\right) \quad (4-18)$$

**Detection edges.** These edges (plotted in green in Fig. 4.12) connect the beginning node  $b_i$  and end node  $e_i$ , with flow  $f_{\text{prob}}(i) = 1$  if  $\mathbf{o}_i$  belongs to  $V_k$ , and 0 otherwise. If all the costs of the edges are positive, the solution to the minimum-cost problem is the trivial null flow. Consequently, we represent each observation  $\mathbf{o}_i$  with two nodes and a detection edge in between with negative cost  $C_{\text{prob}}(i) = \log(1 - P_{\text{prob}}(\mathbf{o}_i))$ . The higher the likelihood (vesselness) of a detection  $P_{\text{prob}}(\mathbf{o}_i)$  the more negative the cost of the detection edge, hence, confident detections are likely to be in the path of the flow in order to minimize the total cost.

**Entrance and exit edges.** Entry edges (purple in Fig. 4.12) connect the source

$S$  with all the end nodes  $e_i$ , with cost  $C_{in}(i) = -\log(0.1) = 1$  and flow  $f_{in}(i)$ . Similarly, exit edges connect the start node  $b_i$  with sink  $T$ , with cost  $C_{out}(i) = -\log(0.1) + C_{prob}(i) = 1 + C_{prob}(i)$ , to ensure that a trajectory ends at a detection with high probability, and flow  $f_{out}(i)$ . The flows are 1 if the trajectory  $V_k$  starts/ends at  $\mathbf{o}_i$ .

#### 4.5.4

##### Iterative vessel branch detection

In order to solve Eq. (4-16), we can take several Linear Programming solvers from the optimization community, such as Simplex or k-shortest paths (Dantzig63). Nonetheless, these solvers would find many trajectories representing the same vessel since there are many nodes (detections) inside a single vessel. To tackle this problem, we propose an iterative procedure which is both fast and still finds the global solution for each of the vessels consecutively.

The first vessel  $V_1$  is found by solving Eq. (4-16), allowing entering flow only at the seed point and setting the maximum flow going out of node  $t$  to be 1. Afterwards, we impose the fact that if a vessel's path is not distant enough from  $V_1$ , it cannot be included in the set of solution paths, avoiding the creation of multiple paths representing the same vessel. To implement this in the graph structure we propose an edge connecting vessel detections in the same layer with a penalty cost, hereby called "toll", which is represented by the thick black edge in Fig. 4.12. This "toll" cost is defined by:

$$C_{toll}(i, j) = K_{toll} \cdot \exp\left(\frac{-\|(\mathbf{o}_i - \mathbf{o}_j)\|}{D_{radius}}\right) \quad (4-19)$$

Therefore, for all points  $\mathbf{o}_j$  which are at a distance  $D_{radius}$  or less than any point of the previously found vessel  $V_1$ , we compute the corresponding  $C_{toll}$ , which will be added to Eq. (4-16). With this we ensure the new path found will be different enough from  $V_1$ , since the summation of costs through a very similar path would be heavily penalized.

For detecting bifurcations the trick is quite simple. We update the flow conditions, allowing any point of  $V_1$  to be the start of a new vessel as shown in Fig. 4.13. Formally this is done by allowing  $f_{in}$  to range between 0,1 in such vessel detection points, being defined as zero in all the other points. Thus we inherently deal with bifurcations, and new vessels are found until the cost of finding a new vessel is no longer negative. We set  $K_{toll} = 5$  for all experiments, empirically.

An overview of the proposed optimization for both detection and tracking is shown in Algorithm 2.

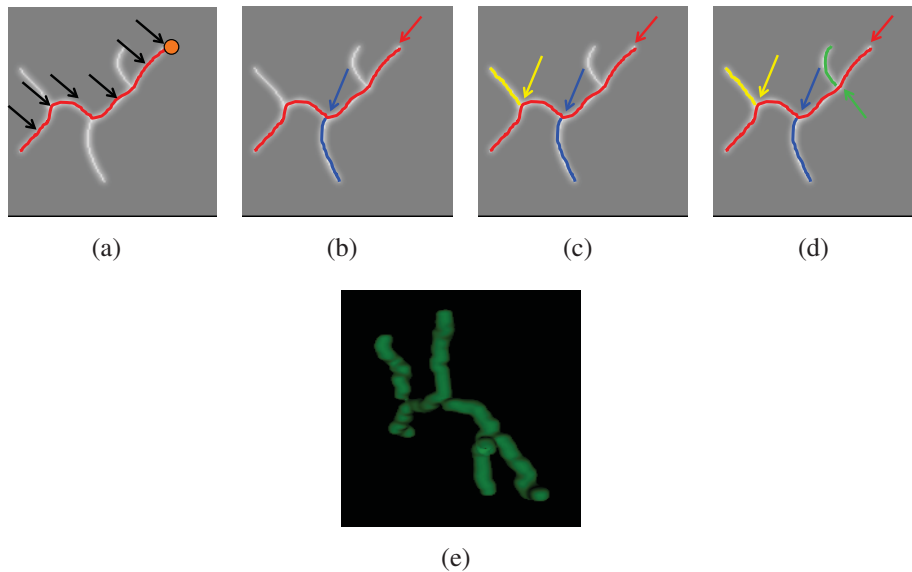


Figure 4.13: Proposed optimization method example of synthetic images. (a) Initial point marked in orange. First path found is the one with minimum cost (red). In the next step, sources will be added along the path (black arrows). (b,c,d) Paths found iteratively, sources of each path marked by colored arrows. (e) 3D view of the vascular network.

## 4.6

### Vascular Branches Validation

Once we detected connected paths from the vascular network, as explained in previous section, a validation procedure is performed to ensure that the final outcome follows some basic rules of vascular anatomy.

These rules are listed bellow and depicted in figure 4.14:

1. Vessel branches should not be too close from each other, in which case, the branches detected probably represent the same vessel. This is implemented by computing a minimum distance between the branches found at each iteration. If this value is lower than a threshold, the branch is discarded.
2. Vessel branches should not reconnect to the already segmented vascular network, so as to avoid loops, which do not exist in vascular networks. This is implemented by computing the minimum distance between the branches found and the already segmented network. If this value is lower than a threshold, the branch forming a loop is discarded.

The threshold cited above is a parameter set as a proportion to the vessel radius size estimated at that point. In this way it is changed dynamically during the segmentation process, so as to both avoid misconceptions and ensure that small vessels detection will not be hindered by the post processing procedure.

**Algorithm 2** Iterative vessel network tracking

---

```

while  $\|\mathcal{S}\| > 0$  do
  1. Get a seed from  $\mathcal{S}$  and find the vessel direction
  2. Compute a sampling cloud of points as shown in section 4.3
  3. Build the graph from the sample points
  4. Compute vesselness measurements at sample points given Eq. (4-13)
  while  $C(V_n) < 0$  do
    5. Find vessel  $V_n$  with minimum cost
    6. Compute the toll charges (Eq. (4-19)) and new flow conditions.
  end while
  6. Define new seeds  $\mathcal{S}$ 
end while

```

---

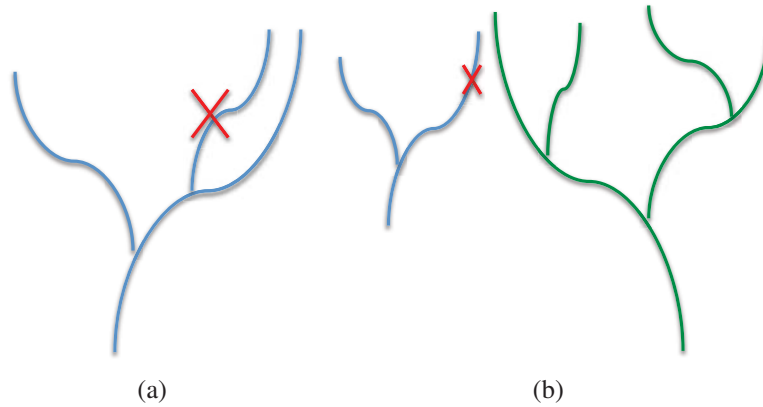


Figure 4.14: Post processing rules implement some anatomical constraints for a vascular network. The red 'X' shows the branches which would be eliminated following each rule (a) and (b) described in section 4.6.

**4.6.1****Next seeds definition**

The final step is to define new seeds to be used as start points for new conical samplings and vascular network detections in the iterative methodology. Each branch found and validated derives a new seed, and to add robustness to the process, a simple procedure was defined to ensure the best seed at each branch is found. A simple search procedure is implemented considering the ending part (30% last points) of each branch found and the node with the best vesselness value is defined as the new seed, as depicted in figure 4.15.

Each new seed defined starts a new segmentation process, which iteratively (recall figure 4.4) implements the segmentation of a full vascular network.

The next chapter shows some results, and discusses the experimental analysis.

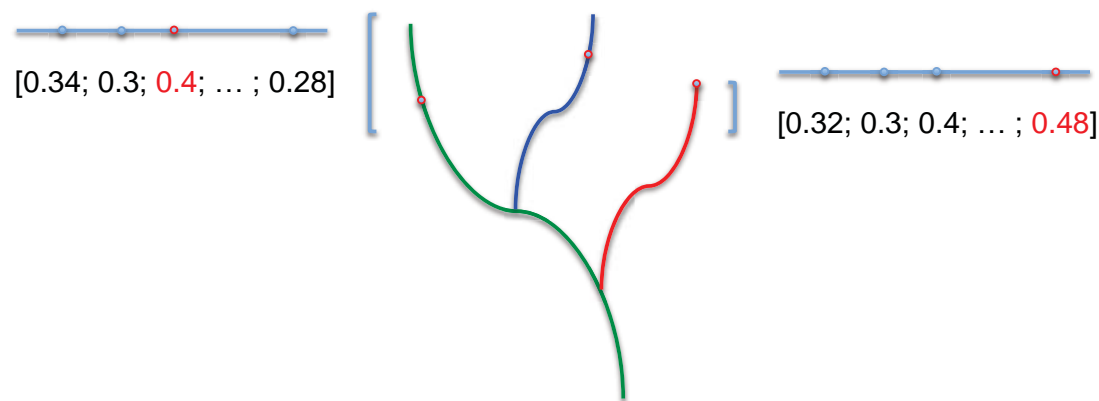


Figure 4.15: Next seeds definition.

## 5

# Results and Experimental Analysis

This chapter provides results and experimental analysis. Also presented the software prototypes developed and the quantitative and qualitative evaluations of the methodology are also presented. Cases of poor performance are discussed to indicate further research.

### 5.1

## Software Prototypes

To implement the ideas proposed in this thesis two prototypes were developed. One runs in batch mode for analyzing a great number of exams sequentially. The other implements a visual interface (GUI) with tools for editing, visualizing and completing the segmented vascular network. They were implemented using C++ and Qt, VTK (Schroeder98) and ITK (Yoo02) libraries.

#### 5.1.1

### Batch software

The batch software runs at prompt command line with the following parameters:

- vesseltrack - identifier for our algorithm inside the executable (there are few other classic methods and batch modes implemented)
- {path to image} - path to the image in MHD format
- {output path} - txt file where the centerline points are written
- {initial radius} - estimated vessel radius at the given initial point
- {sensitivity factor} - value in  $[0,1]$  range that defines how sensitive the vessel point detector will be. The lower this value the more points one gets (and more noise).
- {angle} - opening angle from the cone model used on conical sampling
- {axis size} - axis size of the cone model used on conical sampling
- {sampling distance} - physical distance in mm between the sampled points
- $\{D_{\text{radius}}\}$  - from equation (4-19), this distance that implements the cost penalty to avoid creation of parallel paths on network mapping module.

- $\{D_{\max}\}$  - from equation (4-18), this value that allows gaps on paths found on network mapping module. The higher this value the bigger is the allowed gap.
- $\{\text{max chains}\}$  - max number of chains found on the network mapping module. 0 means no restriction.
- $\{x \text{ initial}\}$  - initial point x coordinate
- $\{y \text{ initial}\}$  - initial point y coordinate
- $\{z \text{ initial}\}$  - initial point z coordinate

It has also a special configuration setup for looping different parameters, where the user defines the ranges and size of the iterative step for each parameter. For looping through different exams bash scripting was used.

### 5.1.2

#### GUI software

The GUI software allows for viewing, editing and completing the segmented vascular network. For visualization of results and image conversion we used the library VTK, and for some auxiliary functions of image processing we used the ITK library. The prototype reads DICOM and RAW/MHD images. It is an improvement of the GUI software developed for liver segmentation in (Oliveira09).

The software prototype has three well-defined spaces, as shown in figure 5.1: a control panel, a display area, and an auxiliary panel, where the user can navigate through slices, change contrast and see some image statistics like histograms.

The control panel consists of four different modules: segmentation, editing, modeling (for 3D visualization of structures) and classification. The classification module was not used in this thesis, and therefore not presented. It implements computation of features and basic classification methods based on decision rules, for classifying the segmented structures.

The segmentation module is where the different segmentation algorithms are available. Moreover, in this space the user defines the parameters for segmenting vascular networks using the methodology proposed in this thesis. Figure 5.2-a shows this model highlighted.

In the visualization module the user can change objects colors and opacities, as well as generate their three-dimensional models. A schematic tree of the structures segmented can be defined by the user and allows one to select structures and then generate their 3D models, change their colors and opacities, or monitor their computed features. Figure 5.2-b depicts this module.

The edition module, illustrated in figure 5.2-c, allows the user to edit the results obtained in the segmentation process, correcting imperfections, or even perform fully manual segmentation, which can serve as a reference definition tool



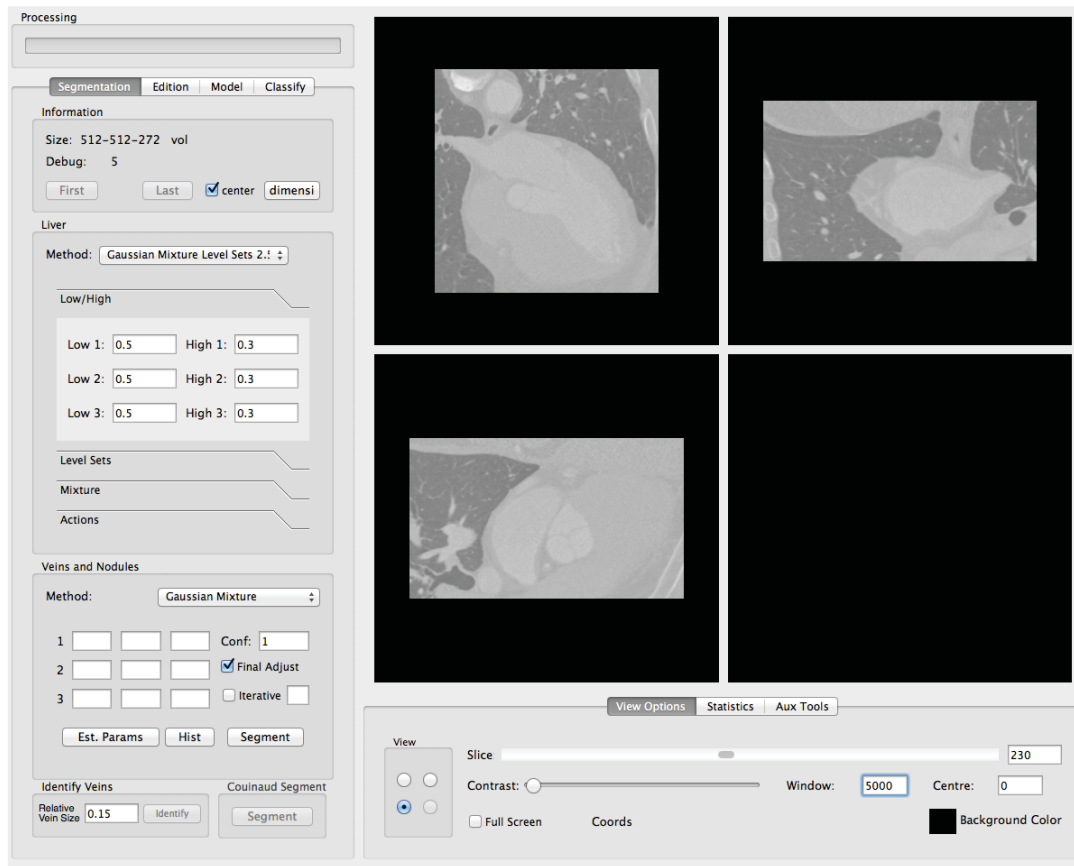


Figure 5.1: GUI software - Main screen

for a specialist. The available edition tools comprise a brush, with which the user can iteratively paint (or erase) pixels; an editor of polygons where the user draws polygons to edit segments; a plane definition to segment large areas; and also some semi automatic tools for completing vascular networks, such as Dijkstra shortest path algorithm.

The display area has four view windows, or a full screen. Each of them is for a specific view: axial, coronal or sagittal, and three-dimensional model of the segmented structures. They can be interactively manipulated with tools that implement zoom, pan, or adjust image parameters.

The bottom auxiliary panel provides the user interaction with the display area for navigating through the slices or adjust contrast, and allow also the inspection of voxel intensity values and coordinates. It is also possible to generate histograms of images and segmented structures.

This tool is meant to allow physicians to actually use the techniques proposed in this thesis as an auxiliary tool for helping in diagnosis or surgical planning.

## 5.2

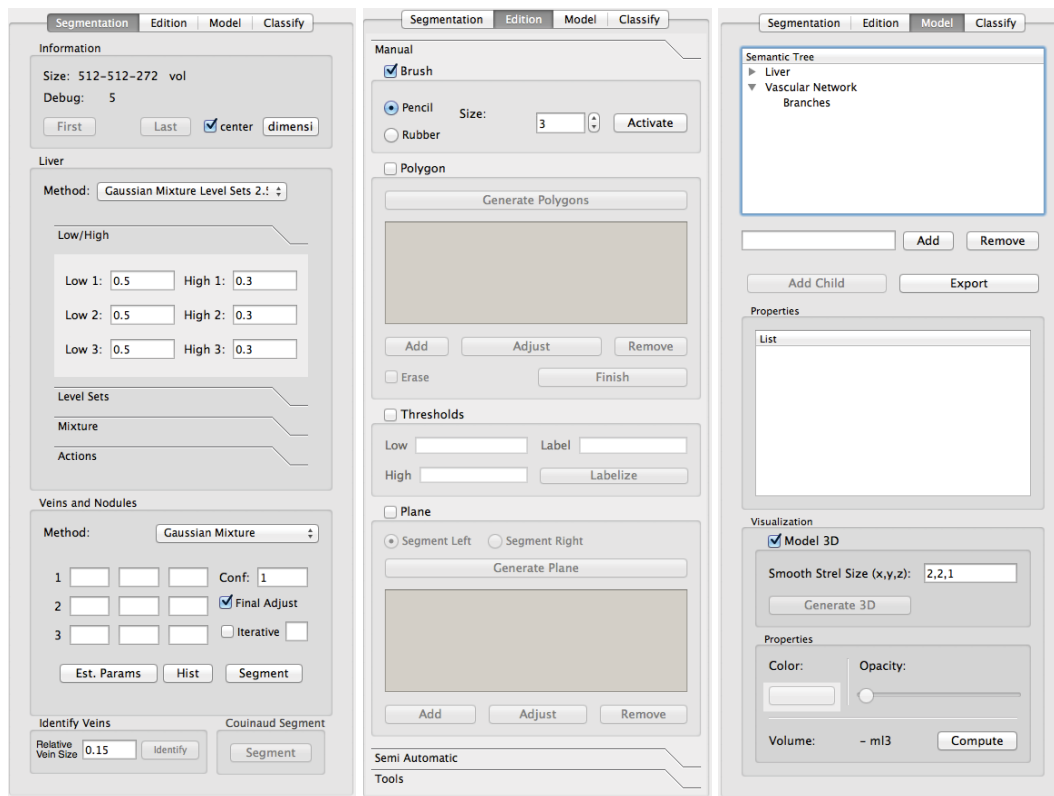


Figure 5.2: GUI Software - Segmentation, Editing and Visualization modules.

## Segmentation Results Evaluation

In this section the results obtained using the proposed methodology are presented. Different datasets were used, and whenever possible the evaluation was done quantitatively. However, manual vascular networks segmentation is a complex and really time demanding task, and therefore usually there is no reference for this kind of anatomical structure. In this cases, the evaluation was qualitative through visual inspection. The use of synthetic data is an alternative to evaluate performance in these cases as well.

Whenever a reference was available, we use accuracy measures following the competition tool made available in the competition website. The measures from which the scores are computed are:

- Root mean squared (RMS) distance between reference and segmented 3D surfaces.
- Hausdorff distance between reference and segmented 3D surfaces.

It is important to mention that the processing time is not easy to measure due to the iterative nature of the methodology, which can take very variable amount of time to run depending on the complexity of the segmented network. Just to give an idea of processing time, an iteration with 5000 points (the maximum amount of points we process) can take between 5 and 10 minutes approximately. Depending

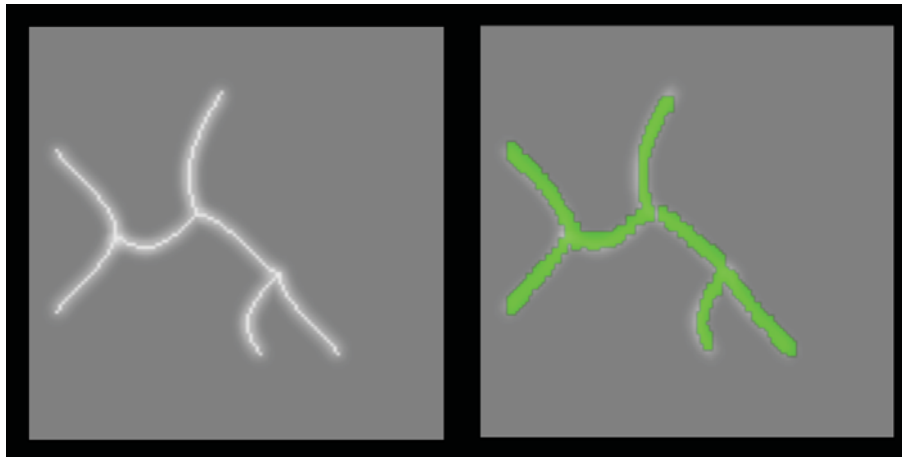


Figure 5.3: Synthetic data segmentation using sinusoidal shaped vascular-like structures

on the complexity of a network, the number of cones needed in the iterative process to segment the whole network vary a lot. It is possible to say that the segmentation of a complex network takes usually up to few hours.

### 5.2.1

#### Synthetic data

The use of synthetic data is very common for evaluating vessel segmentation algorithms. This is mainly due to the difficulty of finding good datasets with references. There are some for single vessel segmentation, but in the case of this thesis, where the object to be segmented is a full vascular network, this is even worse.

Two different synthetic datasets were used. The first one created by (Macedo10) is a planar vascular-like structure modeled using sinusoidal shapes with bifurcations. Additionally the data has also some noise with Gaussian distribution. The result obtained with this dataset was very satisfactory. It is possible to see in figure 5.3 that the vascular network was fully segmented, and even though we had no reference, the visual assessment is enough to ensure that the method succeed in this case.

The second one is more realistic. It generates three-dimensional synthetic blood vessels using stochastic Lindenmayer systems (L-systems) and creating grammars that represent blood vessel architectures. For details refer to (Galarreta-Valverde12, Galarreta-Valverde13). 9 different sequences were used, each of them with different characteristics. The proposed methodology was tested using the whole dataset, achieving maximum accuracies ranging from 99% to 100%, which is an evidence of the potential of the proposed method to track arbitrary network-like structures.

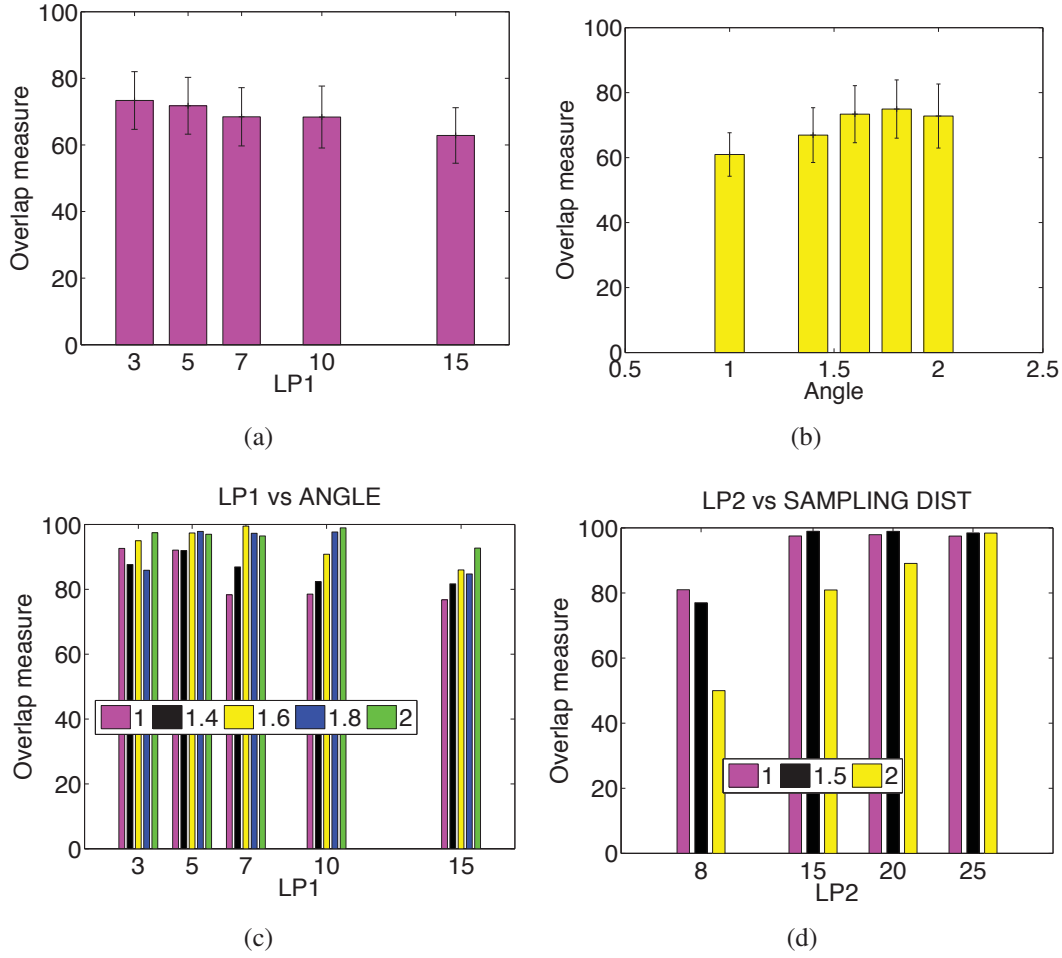


Figure 5.4: Parameter study of  $LP1=D_{\text{radius}}$ ,  $LP2=D_{\text{max}}$ , angle in radians and sampling distance. (see sections 4.3 and 4.5)

To assess the sensitivity of the parameters involved in the process, we used this controlled dataset to test the main parameters and get an useful insight of how the algorithm works under stress. The results obtained are presented in figure 5.4.

The first two parameters tested were the sampling distance and aperture angle of the conical model used for sampling, as described in section 4.3. The other two parameters are related to the Linear Programming optimization described in section 4.5, hereby referred as LP1 and LP2, which correspond to  $D_{\text{radius}}$  from equation (4-19) and  $D_{\text{max}}$  from equation (4-18). In order to compare the results, we computed a simple overlap measurement between the centerline found during the segmentation process and the one of the original image, which corresponds to the centerline of the vessel network.

In figure 5.4(a), we analyze the effect of parameter LP1. The average overlap measures of all 9 sequences is shown and it is possible to notice that on average LP1 must be kept small, otherwise bifurcations are not detected because toll costs are too high, causing the drop in accuracy. In figure 5.4(b) we make a similar study for the Angle parameter, where it is possible to see that the best results are achieved

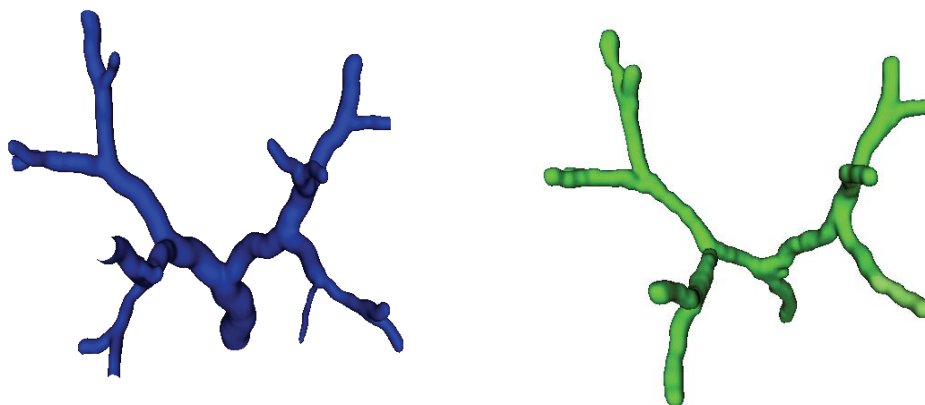


Figure 5.5: Segmentation using synthetic data generated by L-Systems

with an angle of 1.8 radians. If the angle is too small bifurcations cannot be properly followed because they do not fall inside the cone. On the other hand, if the angle is too large, there is too much noise in the cone and spurious paths can be found.

In figure 5.4(c) we see the relationship between LP1 and Angle. As the angle value used gets higher, the LP1 value which gives best results is also increased, otherwise too many false bifurcations could be found. Nonetheless, we still see the results are very stable with a wide range of parameters configuration. Finally, figure 5.4(d) shows a plot of LP2 and Sampling distance, where it is depicted a direct relation between sampling distance and LP2, i.e., high values for sampling distance need high values of LP2 to deliver good results. This is expected, since  $C_t$  values are related to the distance between nodes, which are clearly affected by the sampling distance. A sampling distance of 1-1.5 mm gives very good and stable results for the experiments performed.

Visually it is also possible to assess the quality of the outcome generated with the vascular-like networks being generally found, as shown in figures 5.6 and 5.5.

### 5.2.2 Pulmonary Data

This dataset concerns the pulmonary vessels found inside the lungs. The dataset used was made available by the Extraction of Airways from CT 2009 (EXACT09) challenge (Lo12). There is a reference available for the pulmonary airways but not for the pulmonary vascular network, therefore, our evaluation for this case is qualitative. The goal of this challenge is to compare the results of various algorithms to extract the airway tree from chest CT scans using a common dataset and performance evaluation method. The training and testing datasets of this study

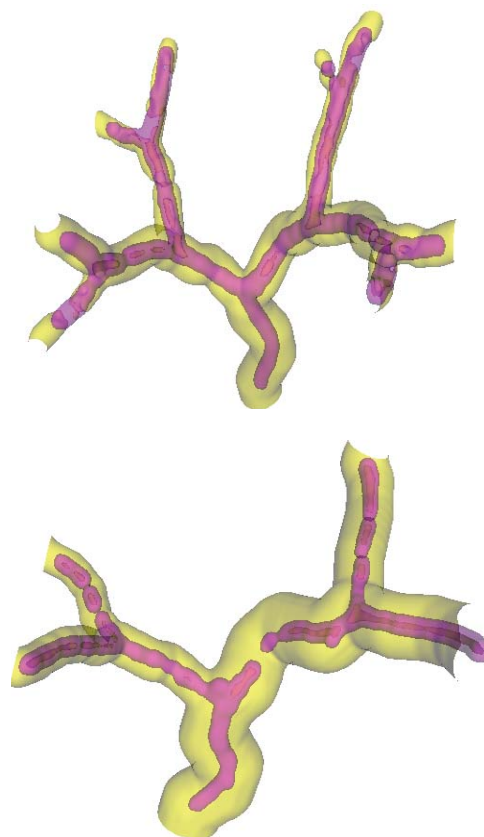


Figure 5.6: Synthetic data segmentation using L-Systems

are available for download.

It is possible to verify visually in figure 5.7 that a great part of the vascular network including many of the bifurcations is found. Even though this kind of evaluation is not really reliable, it shows the potential of the methodology, specially if one takes into consideration that a single seed point was used.

### 5.2.3

#### Coronary Data

A very important application for vascular segmentation concerns the cardiac vessels. To evaluate our algorithm in this case, the coronary dataset (Schaap09) was collected for the MICCAI 2008 workshop "3D Segmentation in the Clinic: A Grand Challenge II" at the 11th MICCAI that took place in September 2008. It is a database containing thirty-two cardiac CTA datasets with reference standard available for the four main coronary vessels. It is important to notice that the reference is composed by four single vessel segmentations, and therefore is not exactly suited to our problem since they do not form a full vascular network for the heart. Nonetheless, it is a very interesting dataset and the available references can be used as guidance for visual assessment.

It is possible to see in figure 5.8 that the reference single vessel is among the

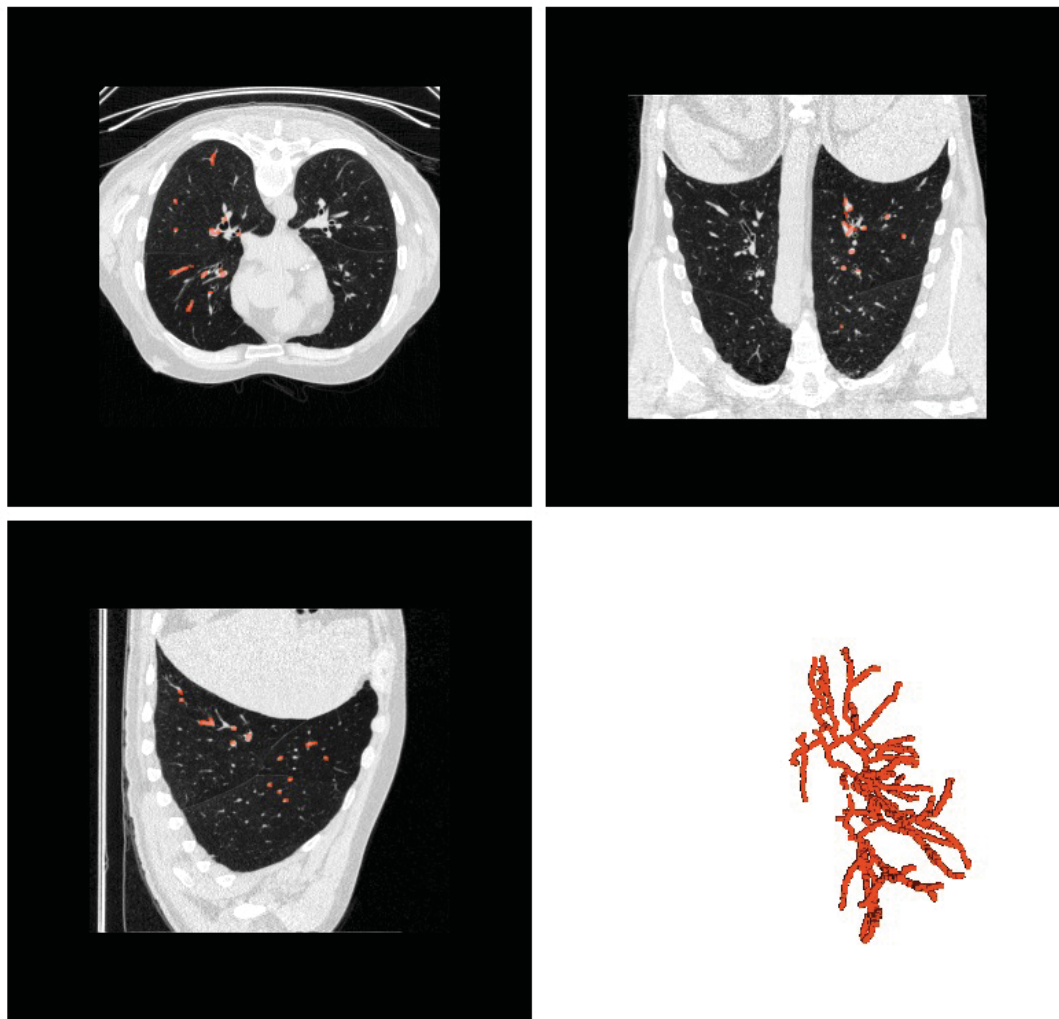


Figure 5.7: Vascular segmentation for pulmonary real dataset

branches segmented on the vascular network derived by the proposed methodology. Other two branches are also segmented.

Considering the single vessel reference available for this dataset, an interesting effect is noticed. Since we use a single start point for segmenting the vascular network, a "blind effect" is observed. Even though the segmentation process segments the network and includes the reference single vessel, it also segments other branches, and therefore would be badly evaluated. If we restrict the algorithm to find a single path, then it is not possible to ensure a priori which of the three detected branches will be chosen. Figure 5.9 depicts the described effect.

#### 5.2.4

##### Liver Data

Another application for vascular segmentation is to segment the hepatic veins for surgical planning. The liver dataset tested here was taken from the Segmentation of the Liver Competition 2007 (SLIVER07) website (Heimann07).



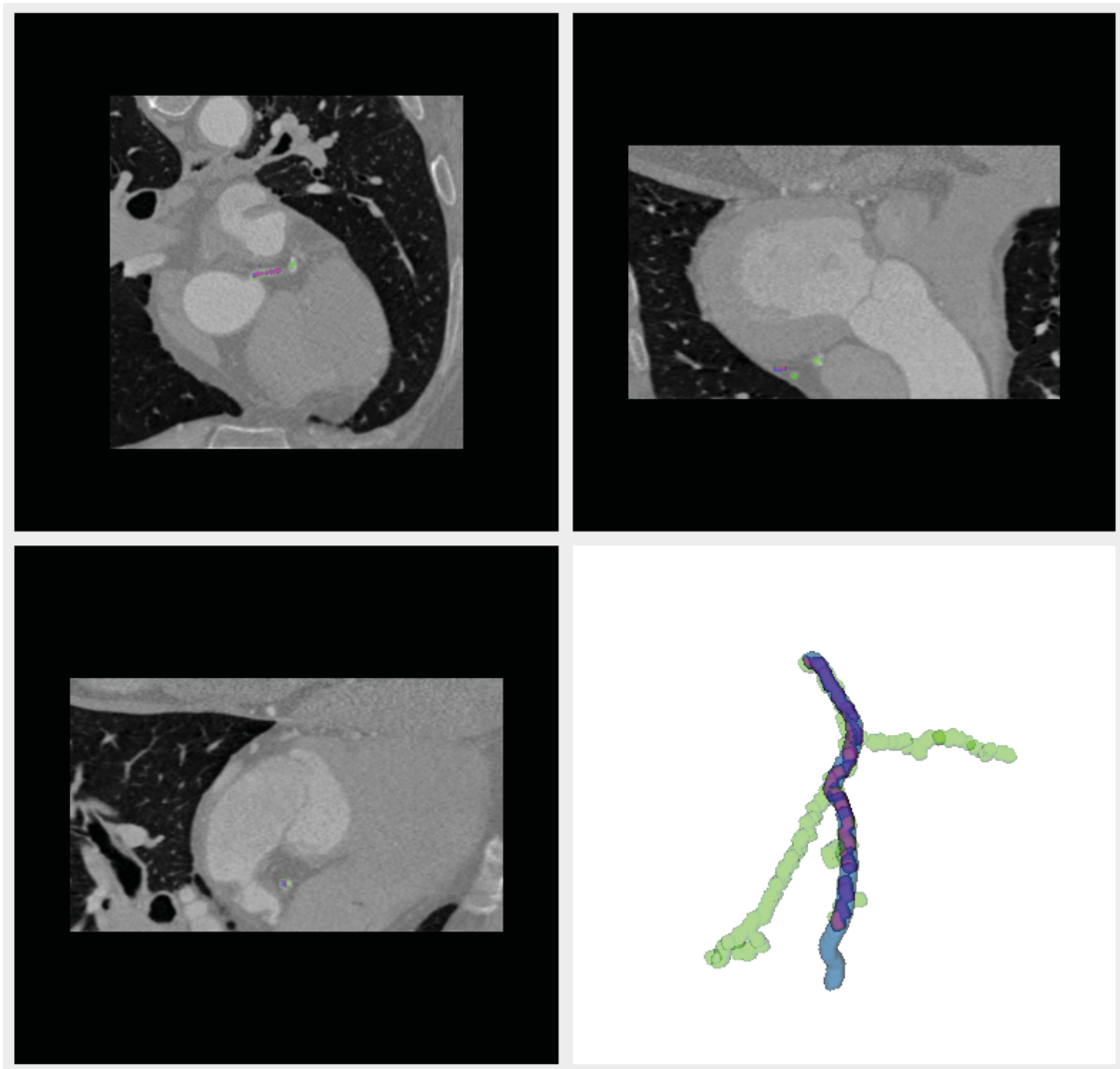


Figure 5.8: Vascular segmentation for coronary real dataset

The goal of that competition was to compare different algorithms to segment the liver from clinical 3D CT scans.

Only the reference data for the liver delineation is available, and the hepatic vascular network is not identified. Therefore, the evaluation is also qualitative in this case. Since the anatomy of vessels inside the liver is somewhat well behaved, we placed a single seed point at the root of the left hepatic vein, and visually verified the segmentation of this branch of the hepatic vascular system, as figure 5.10 shows.

### 5.2.5 Carotids Data

Carotids are the vessels that irrigate the brain, and therefore are very important for many medical conditions such as the cerebrovascular accident, commonly known as stroke. We also tested our algorithm in segmenting the carotid vessels, and the dataset used (Hameeteman11) was created for the 3rd MICCAI Workshop



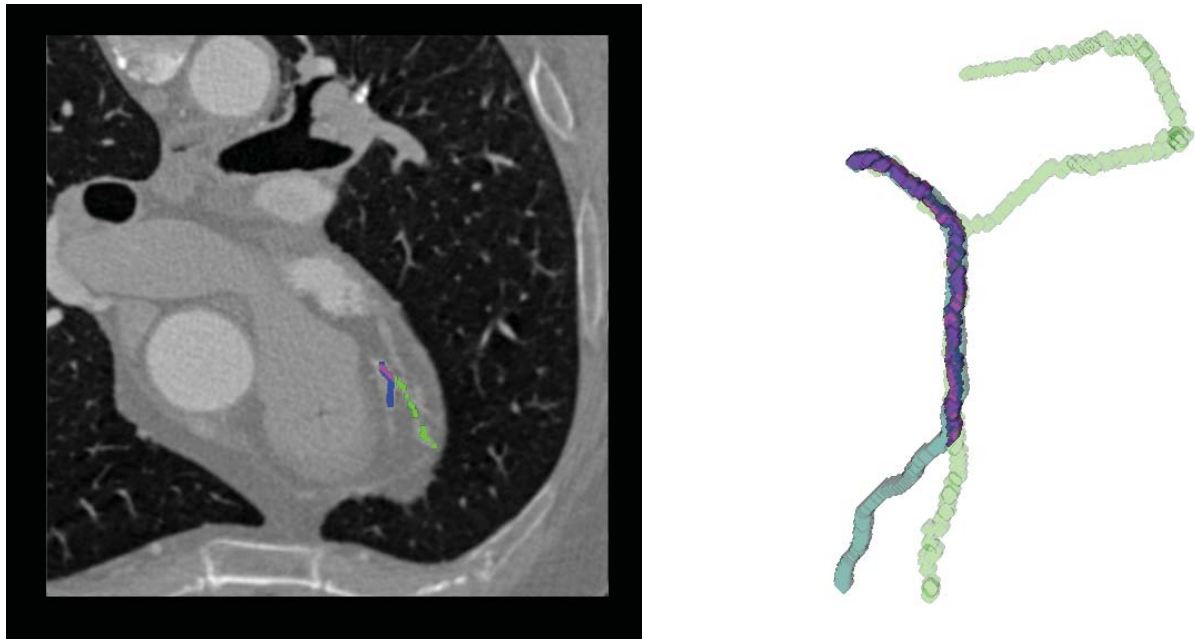


Figure 5.9: This figure shows what we called the blind effect. It is possible to visually understand that since our algorithm does not take into account a specific vessel end point to follow a vessel path, it will not necessarily find a desired vessel, but segment all the vessels connected to the given start point.

in the series "3D Segmentation in the Clinic: a Grand Challenge III", which was held on September 2009 at the 12th MICCAI.

The reference available here concerns the identification of carotid bifurcation. Even though the carotid vascular network reference is not available, it is possible to assess visually the outcome, since these vessels are morphologically simple, usually having a single main bifurcation. It is possible to see in figure 5.11 that both vessels and the bifurcation are found.

### 5.2.6

#### Olfactory Projection Fibers (OPF)

This dataset is actually not from a vascular system, but from nervous fibers of the olfactory system. This, of course, hinders the detection of the vascular network, albeit good results can be achieved. The dataset is available at the DIADEM (short for Digital Reconstruction of Axonal and Dendritic Morphology) challenge website (Brown11) that was a competition for evaluating algorithmic methods for automated neuronal tracing.

OPF are network-like structures, but the inner part, corresponding to the lumen in vessels, are not exactly homogeneous and therefore the Gaussian mixture model proposed for vesselness evaluation, derives low likelihood values. Still, it is

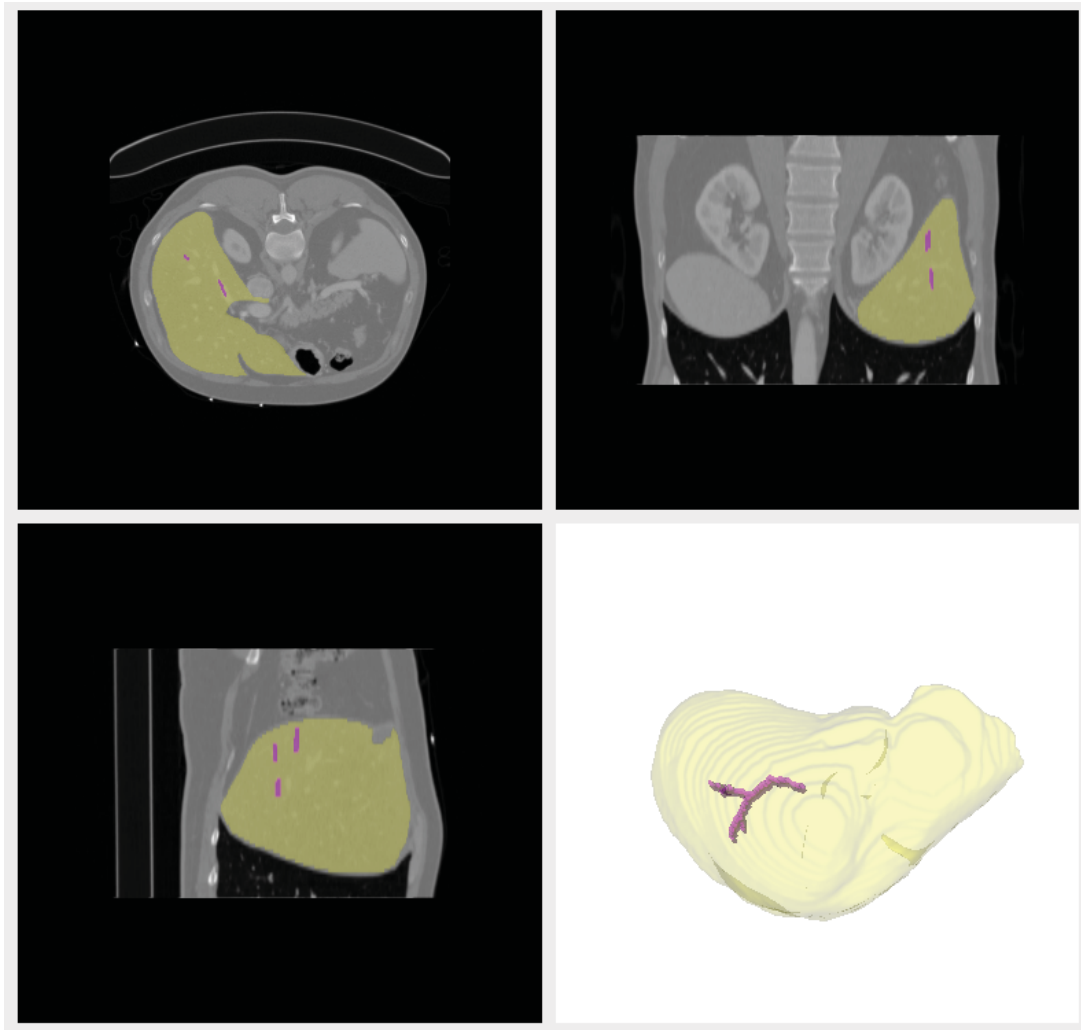


Figure 5.10: Vascular segmentation for liver real dataset

possible to take advantage of the structure and segment at least part of the network. The best thing about this dataset is that there is a reference available, and therefore it is possible to evaluate our methodology quantitatively as table 5.1 shows, also comparing with other works (Türetken11, Türetken12). Visually figure 5.12 depicts the result obtained in one of the datasets available.

### 5.3

#### Topological Description

The methodology described in this thesis delivers not only the segmentation but also the vascular network topology. It is possible to detect vascular branches and identify connecting points, which can be assumed to be bifurcations (even though the exact point of a bifurcation is not determined).

It is not possible to evaluate numerically the results for topology with the datasets used, but some visual feedback is given in figures 5.13 and 5.14. It is possible to visualize branches in different colours and therefore assess the vascular

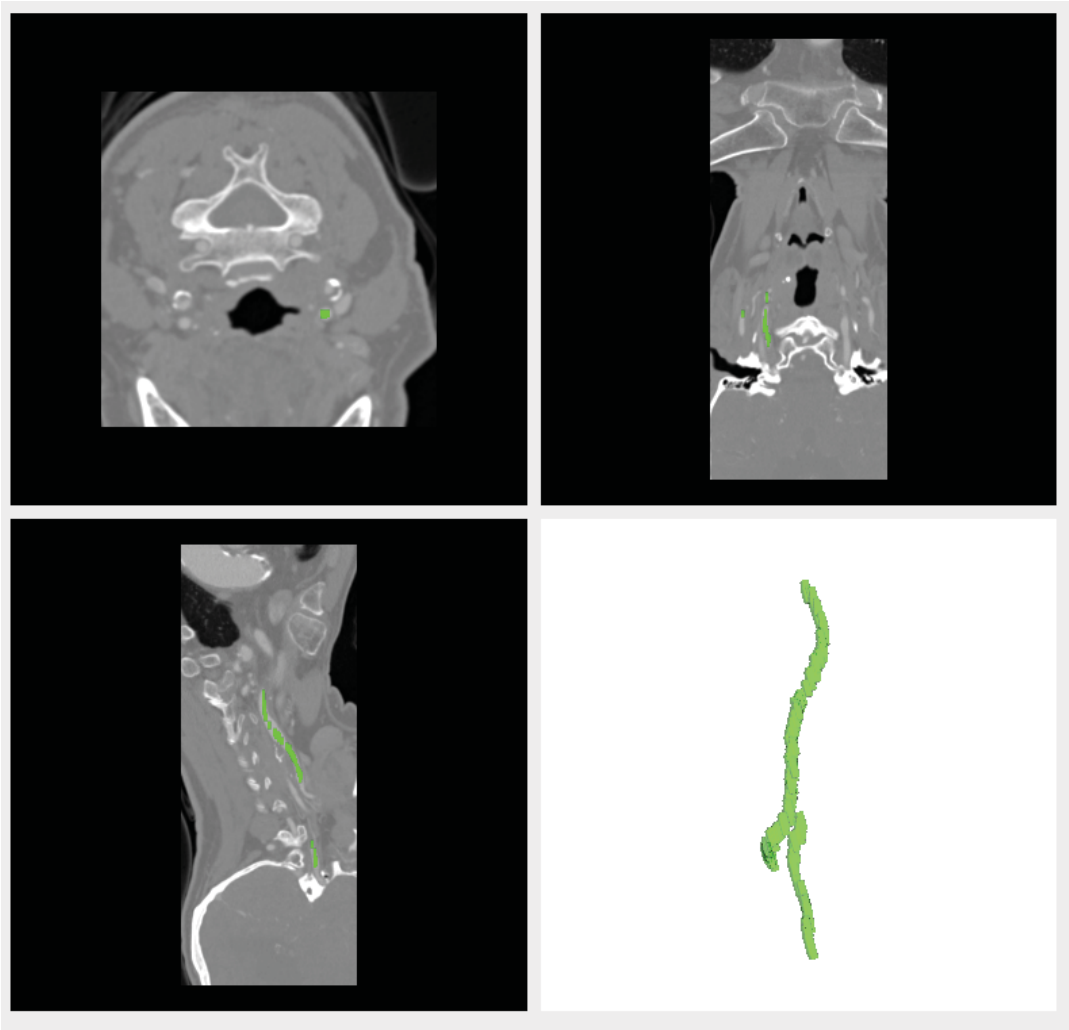


Figure 5.11: Vascular segmentation for carotids real dataset

	Exam1	Exam3	Exam4	Exam5	Exam6	Exam7	Exam8	Exam9
k-MST	–	–	0.865	–	0.898	–	0.722	–
Türetken11								
HGD-QMIP	–	–	0.923	–	0.911	–	0.722	–
Türetken12								
Proposed method	0.800	0.818	0.745		0.833	0.843	0.692	0.327

Table 5.1: OPF database results. The table shows the results obtained for each exam available in the website. The metric is also the one made available for the competition so as to make it possible a straight comparison with other methods.

network topology. The result is coherent with visual inspection.

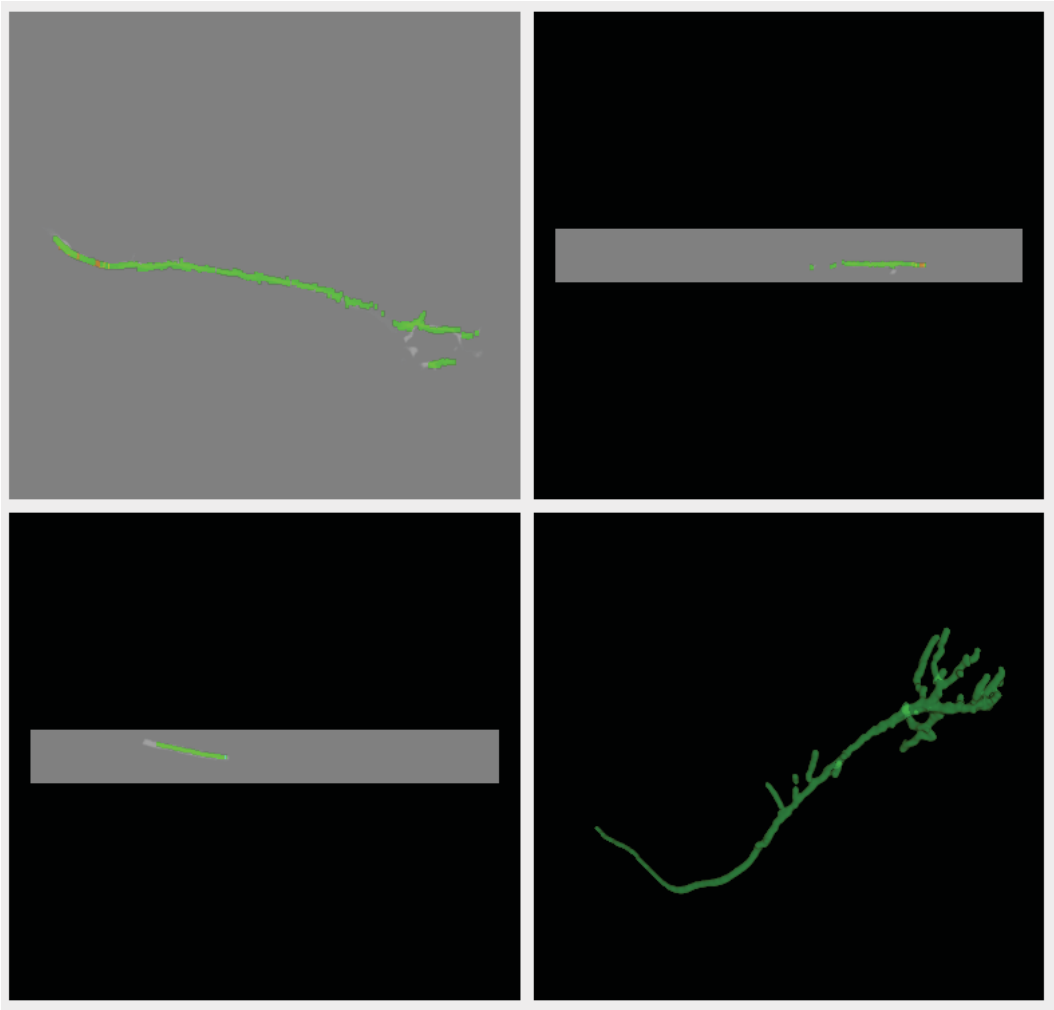


Figure 5.12: Segmentation results for OPF dataset

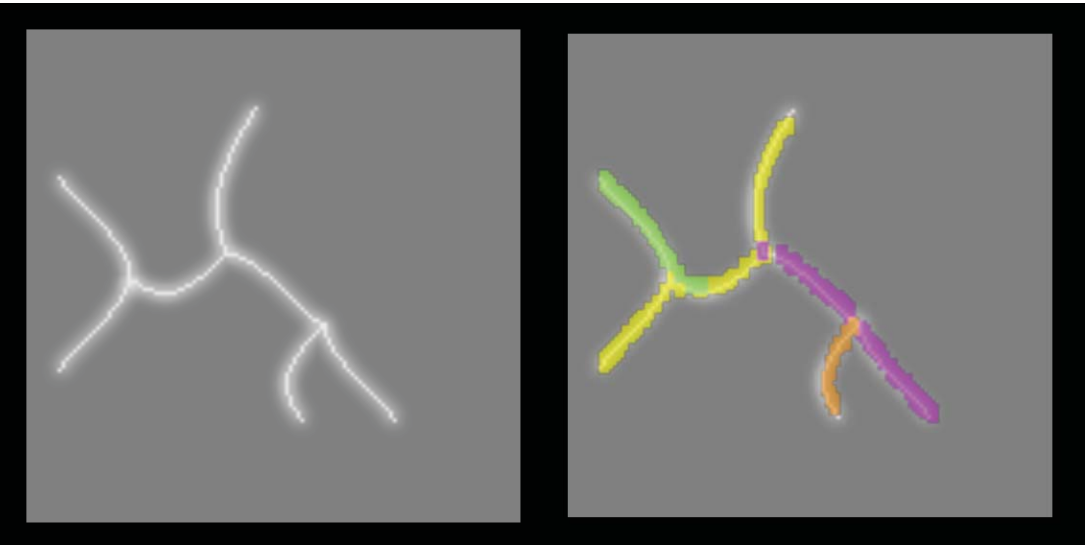


Figure 5.13: Extracted topology for synthetic data

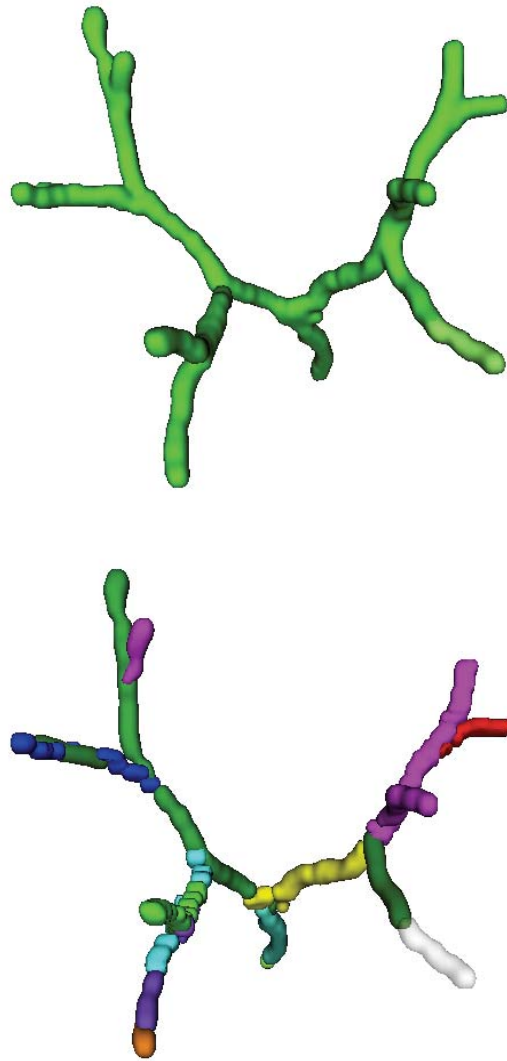


Figure 5.14: Extracted topology for more realistic synthetic data

## 6

### Conclusions

This chapter presents the conclusions of this thesis, summarizing the results obtained and their potential contribution for future applications. It also indicates some possibilities for further research.

#### 6.1

##### Discussion

The methodology presented in this work for the segmentation of vascular networks implements an iterative procedure that segments full vascular networks from a single given seed point. Beginning from that point, a sampling method based on a multilayered spherical cone model defines points to be evaluated for pertaining to a vessel. The points classified as vessel point detections are then organized in a graph structure implemented as a flow network. This network is then analyzed using linear programming in order to find part of a vascular network comprising all branches connected to the seed point reachable in the sampling space. Each branch defines a new seed, and the process restarts from each seed, iteratively segmenting the network until no more branches are found.

The results obtained indicate a good performance, and the different tested datasets attest some robustness. A great problem is the fact that only one dataset was actually evaluated numerically. This is a classical question for vascular networks, since it is very hard to find datasets with segmentation references, mainly due to the difficult and time demanding work involved.

Considering performance, it is possible to conclude that the method derives good results. Synthetic data was segmented without major problems, even when it comprised many bifurcations and vessels with different diameters. The method also produced valid vascular networks in different organs. The pulmonary dataset, with high capillarity, was reasonably segmented, specially if one considers that a single seed point was used. The carotid and coronary datasets stressed the algorithm due to their very hostile environment with many different structures surrounding the vessels, with different textures and densities. Even so, the results obtained were visually comparable with the expected. Finally, the OPF dataset, despite not being a vascular dataset, but actually a nervous fibers network, provided a good

reference, with which we could evaluate numerically the method. It is important to notice that our vesselness measure, derives likelihood for vessels, and therefore it is not completely suited for the task. Still, good results were obtained and the fibers network was segmented properly.

An interesting thing about this work is that it is expandable to other application fields besides medicine. If one changes the vesselness measurement into any other measurement of interest, it is in theory possible to apply the methodology for tracking any network like structure, like road systems and rivers in remote sensing, or particles in high energy physics.

The modular aspect observed in the methodology is also an advantage. It is possible to easily replace the sampling, or the linear programming approach, or the point detection measurement and still take advantage of the remaining modules. This is nice for the continuation of this research, since other state-of-art techniques can be embedded and tested. The next section points some directions for this further research.

## 6.2

### Further Research

Some proposals for further research identified during the development of the thesis, and experimental analysis are proposed here, to spur new developments.

The first one is quite natural. The methodology should be massively tested to assess robustness in very different vascular systems. It should also take advantage of any new dataset that arises with good segmentation references for vascular networks. The visual inspection, even though it gives an idea that things are working fine, is not enough. Numerical analysis can produce much more interesting discussions about extreme cases and how to solve problems observed in those cases.

Another possibility concerns the sampling model. The way it is implemented, the initial definition of parameters for the model is replicated during the whole segmentation process. The problem is that big vessels behave differently from small vessels. Main vessels are usually straighter and longer than secondary smaller vessels, which tend to drift and disperse, changing direction often and suddenly. This could be addressed by dynamically changing the parameters of the conical model, depending on the estimated vessel radius observed at the cone seed point. This could enhance the method in two different ways: it would go faster in main vessels, and would also embed the capacity of segmenting carefully tiny vessels.

Concerning the vesselness measurement and cylinder fit for the estimation of radius and direction of a vessel point, some improvements can also be tried. To begin, the stochastic method used could be replaced by a straight mathematical formulation that derives the same information in a single computation, if possible.

The vesselness model is also based on a cylinder with circular basis, which is not exactly what is observed specially in main vessels with large diameter. These vessels actually are more or less cylindrical but with an ellipsoidal basis, and this could be modelled to assess vesselness better. Of course this means to add one parameter to the optimization process, since one needs two axis to define an ellipse instead of a single radius for the circle.

Finally, an important improvement is related to computational speed. The methodology allows for parallel processing in different stages and could run much faster. For instance, the sampling method computes the vesselness and cylinder optimal parameters for many sample points. They could be all computed at the same time using parallel processes, and this would increase greatly the speed of the segmentation process. Furthermore, every branch found derives a new seed, which is responsible for a new independent iteration. This can also run easily in parallel. These two improvements would have a tremendous impact in the processing time for segmenting networks using the proposed methodology.

### 6.3

#### Final Considerations

Great efforts have been made in the direction of a diagnosis in an acceptable time for the accomplishment of appropriate treatment that ensures longer survival for patients. Much of this development occurs thanks to interaction between professionals and institutions in the areas of technology and the medical field. This cooperation is growing and providing an exchange of knowledge fundamental for the development of tools that aid the physician, generating a direct feedback to society.

This project was designed to promote the development of automated and semi-automatic methods for the segmentation of vascular networks using CT scans. It also fulfilled the objective of providing a software that implements the methodology, allowing the use in diagnosis and surgical planning for different medical conditions.

The methodology provided is meant to be general for different vascular networks and the parameters were also thought for allowing their empirical definition, through the use of specific vascular anatomy knowledge. In this way many different applications can be addressed, such as cardiovascular diseases, cerebrovascular accident inspections or even cancer surgical planning.

The advantages offered by this methodology have been well accepted by the experts involved in the project and therefore further research in this direction is encouraged, including the use of other types of medical imaging beyond the CT, such as MRI.



Moreover, considering the currently available vascular segmentation tools, still very incipient and mostly expensive commercial products, it is reasonable to say that the development of automatic segmentation tools in medical imaging, such as the one presented, is very welcome.

## Bibliography

- [Ahuja93] AHUJA, R.; MAGNANTI, T. ; ORLIN, J.. **Network flows: Theory, algorithms and applications**. Prentice Hall, 1993. 3.3.2, 3.3.2, 3.4, 3.4
- [Bidgood97] BIDGOOD, W.; HORII, S.; PRIOR, F. ; VAN SYCKLE, D.. **Understanding and Using DICOM, the Data Interchange Standard for Biomedical Imaging**. Journal of the American Medical Informatics Association, 4(3):199–212, 1997. 2.2.2
- [Bontrager96] BONTRAGER, K.. **Tratado de Técnica Radiológica e base Anatômica–3<sup>a</sup> Ed.** 1996. 2.2.1
- [Brien94] BRIEN, J.; O'BRIEN, J. F. ; EZQUERRA, N. F.. **Automated segmentation of coronary vessels in angiographic image sequences utilizing temporal, spatial and structural constraints**. Spatial and Structural Constraints, in: Proceedings of SPIE Visualization in Biomedical Computing, p. 25–37, 1994. 1
- [Brown11] BROWN, K. M.; BARRIONUEVO, G.; CANTY, A. J.; PAOLA, V. D.; HIRSCH, J. A.; JEFFERIS, G.; LU, J.; SNIPPE, M.; SUGIHARA, I. ; ASCOLI, G. A.. **The diadem data sets: representative light microscopy images of neuronal morphology to advance automation of digital reconstructions**. Neuroinformatics, 9:143–157, 2011. 5.2.6
- [Camara96] CAMARA, G.; SOUZA, R. C. M.; C[PLEASE INSERT “PRE-RENDERUNICODE–MA” INTO PREAMBLE]RA, G.; CARTAXO, R.; II, F.; SOUZA, M.; FREITAS, U.; GARRIDO, J. ; II, F. M.. **Spring: Integrating remote sensing and gis by object-oriented data modelling**. Computers Graphics, 20:3, 1996. 3.3.1
- [Canny86] CANNY, F. J.. **A Computational Approach to Edge Detection**. IEEE Pattern Analysis and Machine Intelligence, 8(6):679–698, 1986. 3.3.1
- [Canero03] CAÑERO, C.; RADEVA, P.. **Vesselness enhancement diffusion**. Pattern Recognition Letters, 24(16):3141–3151, Dec. 2003. 1

- [Dantzig63] DANTZIG, G.. **Linear programming and extensions**. Princeton University Press, Princeton, NJ, 1963. 3.3.2, 3.3.2, 4.5.4
- [Das11] DAS, S.; SUGANTHAN, P.. **Differential evolution: A survey of the state-of-the-art**. IEEE Transactions on Evolutionary Computation, 15(1):4–31, feb. 2011. 4.4.1
- [Deschamps01] DESCHAMPS, T.; COHEN, L. D.. **Fast extraction of minimal paths in 3d images and applications to virtual endoscopy**. Medical Image Analysis, 5(4):281–299, 2001. 1, 3.3.2
- [Diestel10] DIESTEL, R.. **Graph Theory**. Springer, 2010. 3.2
- [Florin05] FLORIN, C.; PARAGIOS, N. ; WILLIAMS, J.. **Particle filters, a quasi-monte carlo solution for segmentation of coronaries**. MICCAI, p. 246–253, 2005. 1, 4.4.1
- [Friman10] FRIMAN, O.; HINDENNACH, M.; KUEHNEL, C. ; PEITGEN, H.-O.. **Multiple hypothesis template tracking of small 3d vessel structures**. Medical Image Analysis, 14(2):160–171, 2010. 1, 3.3.2, 4.4.1
- [Fujimoto02] FUJIMOTO, H.; GU, L. ; KANEKO, T.. **Recognition of abdominal organs using 3D mathematical morphology**. Systems and Computers in Japan, 33(8):75–83, 2002. 3.3.1
- [Galarreta-Valverde12] GALARRETA-VALVERDE, M. A.. **Three-dimensional synthetic blood vessels generation using stochastic Lindenmayer systems**. Master's thesis, Instituto de Matemática e estatística, Universidade de São Paulo, São Paulo-Brasil, 2012. 5.2.1
- [Galarreta-Valverde13] GALARRETA-VALVERDE, M. A.; MACEDO, M. M. G.; MEKKAOU, C. ; JACKOWSKI, M. P.. **Three-dimensional synthetic blood vessel generation using stochastic L-systems**. SPIE Medical Imaging 2013, 2013 in press. 5.2.1
- [Hameeteman11] HAMEETEMAN, K.; ZULUAGA, M.; FREIMAN, M.; JOSKOWICZ, L.; CUISENAIRE, O.; VALENCIA, L. F.; GLSN, M.; KRISIAN, K.; MILLE, J.; WONG, W.; ORKISZ, M.; TEK, H.; HOYOS, M. H.; BENMANSOUR, F.; CHUNG, A.; ROZIE, S.; VAN GILS, M.; VAN DEN BORNE, L.; SOSNA, J.; BERMAN, P.; COHEN, N.; DOUEK, P.; S[PLEASE INSERT “PRERENDERUNICODE-NC” INTO PREAMBLE]HEZ, I.; AISSAT, M.; SCHAAP, M.; METZ, C.; KRESTIN, G.; VAN DER LUGT, A.; NIESSEN, W. ; VAN WALSUM, T.. **Evaluation**

- framework for carotid bifurcation lumen segmentation and stenosis grading.** *Medical Image Analysis*, 15(4):477 – 488, 2011. Special section on IPMI 2009. 5.2.5
- [Hegadi10] HEGADI, R.; KOP, A. ; HANGARGE, M.. **A survey on deformable model and its applications to medical imaging.** *IJCA, Special Issue on RTIPPR*, (2):64–75, 2010. Published By Foundation of Computer Science. 1
- [Heimann07] T., H.; VAN GINNEKEN B. ; M., S.. **3D Segmentation in the Clinic: A Grand Challenge.** *3D Segmentation in the Clinic: A Grand Challenge*, p. 7–15, 2007. 5.2.4
- [Holash99] HOLASH, J.; MAISONPIERRE, P. C.; COMPTON, D.; BOLAND, P.; ALEXANDER, C. R.; ZAGZAG, D.; YANCOPOULOS, G. D. ; WIEGAND, S. J.. **Vessel cooption, regression, and growth in tumors mediated by angiopoietins and vegf.** *Science*, 284(5422):1994–1998, 1999. 1
- [Kim00] KIM, S.; YOO, S.; KIM, S.; KIM, J. ; PARK, J.. **Segmentation of kidney without using contrast medium on abdominal CT image.** *International Conference on Signal Processing Proceedings, WCCC-ICSP*, 2, 2000. 3.3.1
- [Kirbas04] KIRBAS, C.; QUEK, F. K. H.. **A review of vessel extraction techniques and algorithms.** *ACM Comput. Surv.*, 36(2):81–121, 2004. 1
- [Krissian06] KRISSIAN, K.; AND. **Smooth Vasculature Reconstruction with Circular and Elliptic Cross Sections.** *Proc. of The Medicine Meet Virtual Reality (MMVR 14) conference*, 2006. 1
- [Lamecker05] H., L.; S., Z.; H., H. ; M., S.. **Medical Applications for Statistical 3D Shape Models.** *Proc. Computer Aided Surgery Around the Head*, 17:61, 2005. 3.3.2
- [Lesage09] LESAGE, D.; ANGELINI, E.; BLOCH, I. ; FUNKA-LEA, G.. **A review of 3d vessel lumen segmentation techniques: Models, features and extraction schemes.** *Medical Image Analysis*, 13(6):819–845, 2009. 1
- [Lo12] LO, P.; VAN GINNEKEN, B.; REINHARDT, J. M.; YAVARNA, T.; DE JONG, P. A.; IRVING, B.; FETITA, C.; ORTNER, M.; PINHO, R.; SIJBERS, J.; FEUERSTEIN, M.; FABIJANSKA, A.; BAUER, C.; BEICHEL, R.; MENDOZA, C. S.; WIEMKER, R.; LEE, J.; REEVES, A. P.; BORN, S.; WEINHEIMER, O.; VAN RIKXOORT, E. M.; TSCHIRREN,

- J.; MORI, K.; ODRY, B.; NAIDICH, D. P.; HARTMANN, I.; HOFFMAN, E. A.; PROKOP, M.; PEDERSEN, J. H. ; DE BRUIJNE, M.. **Extraction of airways from ct (exact'09)**. IEEE Transactions on Medical Imaging, 31(11):2093 –2107, nov. 2012. 5.2.2
- [Lorigo01] LORIGO, L. M.; FAUGERAS, O. D.; GRIMSON, W. E. L.; KERIVEN, R.; KIKINIS, R.; NABAVI, A. ; WESTIN, C.-F.. **Curves: Curve evolution for vessel segmentation**. Medical Image Analysis, 5:195–206, 2001. 1
- [Lüscher03] LSCHER, T. F.; CREAGER, M. A.; BECKMAN, J. A. ; COSENTINO, F.. **Diabetes and vascular disease: Pathophysiology, clinical consequences, and medical therapy: Part ii**. Circulation, 108(13):1655–1661, 2003. 1
- [MS04] MINISTÉRIO DA SAÚDE, S. D. V. E. S.. **Uma análise da mortalidade no brasil e regiões**, June 2004. 1
- [Macedo10] MACEDO, M.; MEKKAOU, C. ; JACKOWSKI, M.. **Vessel centerline tracking in cta and mra images using hough transform**. Lecture Notes in Computer Science, 6419:295–302, 2010. 5.2.1
- [Manniesing04] MANNIESING, R.; NIESSEN, W.. **Local speed functions in level set based vessel segmentation**. Medical Image Computing and Computer-Assisted Intervention - MICCAI 2004, 3216:475–482, 2004. 1
- [Masutani06] MASUTANI, Y.; UOZUMI, K.; AKAHANE, M. ; OHTOMO, K.. **Liver CT image processing: A short introduction of the technical elements**. European Journal of Radiology, 58(2):246–251, 2006. 3.3
- [McInerney96] MCINERNEY, T.; TERZOPOULOS, D.. **Deformable models in medical image analysis: a survey**. Medical Image Analysis, 1(2):91 – 108, 1996. 1
- [Olabarriaga03] OLABARRIAGA, S.; BREEUWER, M. ; NIESSEN, W.. **Minimum cost path algorithm for coronary artery central axis tracking in ct images**. Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003, 2879:687–694, 2003. 1, 3.3.2
- [Oliveira09] OLIVEIRA, D. A. B.. **Segmentação e visualização do fígado, seus vasos e sub-regiões a partir de imagens de tomografia computadorizada da região abdominal**. Master's thesis, Departamento de Engenharia Elétrica, Puc-Rio, Rio de Janeiro - Brasil, 2009. 1, 5.1.2

- [Oliveira11] OLIVEIRA, D.; FEITOSA, R. ; CORREIA, M.. **Segmentation of liver, its vessels and lesions from ct images for surgical planning.** BioMedical Engineering OnLine, 10(1):30, 2011. 1
- [Osher03] OSHER, S.; FEDKIW, R.. **Level Set Methods and Dynamic Implicit Surfaces.** Springer, 2003. 3.3.2
- [Rossignac07] ROSSIGNAC, J.; WHITED, B.; SLABAUGH, G.; FANG, T. ; UNAL, G.. **Pearling: 3d interactive extraction of tubular structures from volumetric images.** MICCAI Workshop: Interaction in Medical Image Analysis and Visualization, 2007. 1, 4.4.1
- [Schneider96] SCHNEIDER, U.; PEDRONI, E. ; LOMAX, A.. **The calibration of ct hounsfield units for radiotherapy treatment planning.** Physics in Medicine and Biology, 41(1):111, 1996. 2.2.1
- [Schroeder98] SCHROEDER, W.; MARTIN, K. ; LORENSEN, B.. **The visualization toolkit.** Kitware, Inc., 1998. 5.1
- [Schaap09] SCHAAP, M.; METZ, C.; VAN WALSUM, T.; VAN DER GIESSEN, A. G.; WEUSTINK, A. C.; MOLLET, N.; BAUER, C.; BOGUNOVIC, H.; CASTRO-GONZALEZ, C.; DENG, X.; DIKICI, E.; O'DONNELL, T.; FRENAY, M.; FRIMAN, O.; HOYOS, M. H.; KITSLAAR, P. H.; KRISSIAN, K.; KÜHNEL, C.; LUENGO-OROZ, M. A.; ORKISZ, M.; SMEDBY, Ö.; STYNER, M.; SZYMCAK, A.; TEK, H.; WANG, C.; WARFIELD, S. K.; ZAMBAL, S.; ZHANG, Y.; KRESTIN, G. P. ; NIESSEN, W. J.. **Standardized evaluation methodology and reference database for evaluating coronary artery centerline extraction algorithms.** Medical Image Analysis, 13(5):701–714, 2009. 5.2.3
- [Sethian99] SETHIAN, J.. **Level Set Methods and Fast Marching Methods.** Cambridge, 1999. 3.3.2
- [Silverstein94] ROBERT A. SILVERSTEIN, V. B. S.. **The Circulatory System.** Twenty-First Century Books, 1994. 2.1
- [Soler01] SOLER, L.; DELINGETTE, H.; MALANDAIN, G.; MONTAGNAT, J.; AYACHE, N.; KOEHL, C.; DOURTHE, O.; MALASSAGNE, B.; SMITH, M.; MUTTER, D. ; OTHERS. **Fully Automatic Anatomical, Pathological, and Functional Segmentation from CT Scans for Hepatic Surgery.** Computer Aided Surgery, 6:131–142, 2001. 3.3.2

- [Türetken11] TÜRETKEN, E.; GONZÁLEZ, G.; BLUM, C. ; FUA, P.. **Automated reconstruction of dendritic and axonal trees by global optimization with geometric priors**. *Neuroinformatics*, 9:279–302, 2011. 5.2.6
- [Türetken12] TÜRETKEN, E.; BENMANSOUR, F. ; FUA, P.. **Automated reconstruction of tree structures using path classifiers and mixed integer programming**. *Conference on Computer Vision and Pattern Recognition - CVPR*, 2012. 5.2.6
- [Vanessa10] DE OLIVEIRA CAMPOS, V.. **Segmentação Multicriterio para Detecção de Nódulos Pulmonares em Imagens de Tomografia Computadorizada**. PhD thesis, Departamento de Engenharia Elétrica, Puc-Rio, Rio de Janeiro - Brasil, 2010. 3.3.1
- [WHO08] ORGANIZATION, W. H.. **Ten leading causes of deaths in 2008**, June 2008. 1
- [Wan03] WAN, S.-Y.; HIGGINS, W.. **Symmetric region growing**. *IEEE Transactions on Image Processing*, 12(9):1007 – 1015, sept. 2003. 1
- [Wang12] WANG, X.; HEIMANN, T.; LO, P.; SUMKAUSKAITE, M.; PUDERBACH, M.; DE BRUIJNE, M.; MEINZER, H. ; WEGNER, I.. **Statistical tracking of tree-like tubular structures with efficient branching detection in 3d medical image data**. *Physics in Medicine and Biology*, 57:5325–5342, 2012. 1
- [Wink00] WINK, O.; NIESSEN, W. J. ; VIERGEVER, M. A.. **Fast delineation and visualization of vessels in 3d angiographic images**. *IEEE Transactions on Medical Imaging*, 19(4):337–346, 2000. 1
- [Worz07] WÖRZ, S.; ROHR, K.. **Segmentation and quantification of human vessels using a 3-d cylindrical intensity model**. *IEEE Transactions on Image Processing*, 16(8):1994–2004, 2007. 1, 4.4.1
- [Yoo02] YOO, T.; ACKERMAN, M.; LORENSEN, W.; SCHROEDER, W.; CHALANA, V.; AYLWARD, S.; METAXAS, D. ; WHITAKER, R.. **Engineering and Algorithm Design for an Image Processing API: A Technical Report on ITK-the Insight Toolkit**. *Studies in Health Technology and Informatics*, p. 586–592, 2002. 5.1