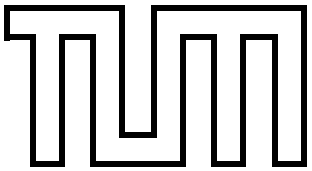


FAKULTÄT FÜR INFORMATIK

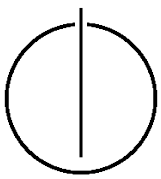
DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Diplomarbeit in Informatik

Automatic Detection of Stent Grafts in Interventional 2D and 3D Image Data

Automatische Erkennung von Stent Grafts
in interventionellen 2D und 3D Bilddaten

Author:	Markus Urban
Supervisor:	Prof. Dr. Nassir Navab
Advisors:	Dipl.-Inf. Univ. Stefanie Demirci, Dipl.-Tech. Math. Univ. Maximilian Baust
Date:	August 15, 2009



I assure the single handed composition of this diploma thesis only supported by declared resources.

Munich, August 15, 2009

Markus Urban

Contents

Preface

Abstract

By now, *endovascular aneurysm repair (EVAR)* has become a well-established medical procedure for the treatment of *abdominal aortic aneurysms (AAAs)*. A *stent graft* is placed inside the aorta using minimally invasive surgery in order to block the blood flow into the aneurysm, which takes the pressure off the diseased tissue and prevents a possible rupture.

Being in use over 10 years already, it is still a very challenging intervention for the surgeon—both in an intra- and postoperative sense—that requires quite an amount of human interaction. Navigation is conducted on basis of interventional 2D images lacking important 3D information, and postoperative control and monitoring involves time-consuming manual segmentation and measurements.

Thus, there is a lot of room for improvement using computer-aided techniques. During the intervention, modern navigation methods are able to incorporate preoperative volumes for advanced visualization techniques guiding the surgeon while reducing the radiation dose for the patient. Standards have to be developed in order to evaluate the quality of the stent graft placement in an objective manner, and to detect stent migration that can occur over time—even years after the implantation—which justifies the need for an easy-to-use, fast and reliable method for postoperative monitoring on a regular basis.

Automatic detection of the stent graft in interventional 2D and 3D image data is the essential basis to achieve these goals, and, thus, constitutes the main concern of this diploma thesis. A *nonrigid model-to-image registration* approach for stent graft segmentation is developed including the necessary methods and a deformable parametric stent segment model inspired by the real geometry and physical properties of the stent graft. The individual segments are registered to the image in a piecewise fashion. The implemented semi-automatic method requires a minimal amount of user interaction for initialization of the first segment, while the other ones are detected automatically. The results are evaluated using phantom data and real patient data.

The focus lies on stent grafts as used for EVAR, which—due to their tubular sine-shaped segments—can easily be described by a mathematical model. However, the proposed approach could be utilized for other parametric stent models, too.

Everything should be made as simple as possible, but not simpler.

–Albert Einstein

Acknowledgments

I have come a long way to the point of completing my degree with this diploma thesis—with just the usual struggle and troubles. I would like to thank everyone who accompanied me on that road trip—most of all the ones who supported and encouraged me constantly.

First to mention personally are my advisors, Stefanie Demirci and Maximilian Baust, for their outstanding assistance including guidance and advice throughout the project as well as constant review and proofreading of the thesis. Their feedback was of tremendous help, and I really appreciate it.

Furthermore, special thanks go to my supervisor, Nassir Navab, who constitutes the major force driving me into the direction of computer aided medical procedures. His admirable way of building up the *Chair for Computer Aided Medical Procedures & Augmented Reality (CAMPAR)* at the *TUM*, as well as his extraordinary motivation in presenting and introducing this field of research along with his profound expertise made it a pleasure to devote myself to the subject. He is also responsible for facilitating my stay abroad in the US for an internship with Siemens Corporate Research, Princeton, which was a great, invaluable experience.

At this point, I would also like to thank all the other people working for the chair for helpful suggestions, and for making the projects I was involved in there throughout my studies a pleasant time.

On the medical side, sincere thanks go to the following departments and hospitals for supplying me with the required image data: *Department of Radiology*, *NTNU-St. Olavs Hospital*, Trondheim, Norway (patient data), and *Nuklearmedizinische Klinik*, *TUM-Klinikum rechts der Isar*, Munich, Germany (phantom data).

Gratitude is especially directed towards the latter, along with the "impersonation" of its CT suite, Thomas Wendler, for his introduction to the device when we were required to scan a stent graft in order to gather phantom data.

Many thanks also go to Dr. med. Mojtaba Sadeghi-Azandaryani (*Surgical Clinic Division for Vascular Surgery and Phlebology*, *LMU-Klinikum Innenstadt*, München, Germany), and Dr. med. Reza Ghotbi (*Vascular Surgery Department*, *Kreisklinik München-Pasing*, München, Germany) for them giving insight into details of the EVAR procedure and answering all our (obscure) questions.

Furthermore, I would like to express my deepest gratitude to the author and maintainer of the data recovery tool GNU `ddrescue`, Antonio Diaz, and everyone who con-

tributed to the program. It really saved my live when both my laptop's main drive and my external backup drive crashed coincidentally—no idea how this could happen, but these things tend to occur at awkward times. Using the tool I was able to restore basically all my thesis-related files (and most of my private data, too)—so thank you, thanks a lot!

Further private thanks—or rather apologies—are directed towards all my friends for not being mad at me although I ignored them completely for a while by not calling, not answering emails, and not responding to invitations. Hereby, I officially apologize for my unacceptable behavior, and I would like to announce that I am on the mend.

Last but not least, it is time to thank my main contributors, my loving parents, for their patience and support—not only financially—on my way to the degree. I would not have been able to make it without them. Just as well, my lovely significant other, Katrin, helped me survive particularly the most straining times of preparing for exams and writing the thesis by providing me with the essential basics in live: Delicious food and a "whole lotta love"!

Outline

Part I: Introduction

Chapter 1: Overview

This chapter provides a general overview and introduction to the fields this thesis is related to.

Chapter 2: Computer Scientific Background

A further introductory chapter approaches the specific topic of this thesis from the computer scientific side. It provides the necessary background on *digital images* and *image processing* in general, and introduces the field of *computer-aided medical procedures* and *medical imaging* as well as *image-guided interventions* in particular.

Chapter 3: Imaging Modalities

Considering the great variety of imaging modalities, just the ones that are relevant for this thesis are explained here. These are *fluoroscopy* and *computed tomography (CT)* including *angiography*.

Chapter 4: Medical Background

Relevant medical terms and procedures are introduced. These include aneurysms and related terms. In addition, general facts and treatment options are outlined. *Endovascular aneurysm repair (EVAR)* is introduced along with details on the workflow consisting of *diagnosis* and *preoperative planning*, the actual *intervention*, and *postoperative follow-up examinations*. Furthermore, the challenges of the intervention for the surgeon are examined along with their meaning for the patient. The role of computer science is pointed out by proposing possible solutions to these issues.

Part II: Problem Statement

Chapter 5: Project Description

The objective and motivation of this thesis is clarified, along with the related work and the contribution provided by this work. The required tasks to realize the project are summarized.

Chapter 6: Problem Analysis and Solution Options

The specific problem is analyzed in detail starting with some general thoughts on addressing it. Furthermore, the a priori knowledge we may be able to utilize is analyzed. Possible solution options via image segmentation methods are introduced and compared, and an overview of the developed approach is presented along with its motivation.

Part III: Methodology

Chapter 7: Fundamentals and Formalizations

This chapter takes care of some fundamental methods and formalizations. It contains formal definitions related to digital images and image segmentation, as well as introductions to registration and optimization including the associated mathematical foundations.

Chapter 8: Methods and Models

The core chapter of this thesis describes and formalizes the developed methods and models. The parametric stent segment model is presented, and the details of the semi-automatic *model-to-image registration* approach are explained. Since the registration problem is solved by *optimization* (using the *downhill simplex method*) the required *objective function* is presented. Furthermore, the utilized *preprocessing* methods (*distance transform* and *background removal*) are outlined.

Part IV: Results and Conclusion

Chapter 9: Experimental Results

The developed approach is tested on phantom data as well as real patient data. Starting from a general analysis of important evaluation criteria for the registration process, the experimental setup is described along with the used test data, the collection of ground truth and the tested parameters. Furthermore, the experimental results are evaluated and discussed.

Chapter 10: Conclusion and Outlook

This chapter discusses the reached goals and open challenges. In addition, it gives an outlook on future work by providing some ideas on improvements. These include advanced preprocessing methods, redesign of the objective function, segmentation-driven registration, and the utilization of more a priori knowledge. Ideas towards full automatization of the registration process conclude this chapter. The closure completes this thesis.

Appendix

Chapter A: Implementation Details

The appendix contains some details on the "VirtualStenting" application that has been implemented for testing purposes. Its usage is explained and illustrated by screenshots. Additionally, the XML model specification is described, which facilitates storing the stent graft properties required for the registration process. The template file is included in order to exemplify its usage.

Part I.

Introduction

1. Overview

Medical imaging has come a long way since the discovery of the capabilities of X-rays by Wilhelm Conrad Röntgen in 1895. It is now at a stage where various different imaging devices are used at a daily basis in hospitals around the world. They have become an essential part of the clinical routine during all stages of the medical workflow like *diagnosis*, *preoperative planning*, *intraoperative navigation* and *postoperative monitoring and treatment*.

Nowadays, all these processes are unfeasible without the help of computers: The huge amount of image data collected during diagnosis and treatment requires intense processing in order to *visualize* as well as *analyze* the data in a way it is useful for physicians. Computers can be a valuable tool for planning and assistance and may save a lot of time by eliminating some manual processes. Thus, a wide field of research in computer science arose in the last decades that is concerned with providing the necessary algorithms for medical image processing. The list of methods and applications includes—but is not restricted to—the following subjects:

Image segmentation is concerned with the extraction (or *segmentation*) of features or objects visible in the image. In the medical domain the latter may be organs (such as liver, heart, etc.), bones (like vertebrae), tumors, and also medical instruments (e.g. catheters, stents, etc.). It may help the surgeon plan, carry out and evaluate the intervention.

Image registration tries to fuse (or *register*) images of (usually) different imaging modalities into a common coordinate frame and to overlay them in order to make use of complementary information the images may contain.

In-situ visualization of medical images can enhance the surgeon's view on and *into* the patient using special display devices (like HMDs, semi-transparent mirrors, etc.). Key techniques are *augmented reality* and advanced *graphics card programming* in conjunction with the study of *perception*, *visualization techniques* and *user interfaces*.

Patient/instrument tracking is required for in-situ visualization in order to show pre- and intraoperative image data as well as virtual instruments at the proper location overlaid with the patient's body.

Image-guided interventions make use of the above techniques to help the surgeon *navigate* through the patient's body with wires, catheters, needles or other devices facilitating *minimally invasive surgery* and other minimally invasive image-guided

procedures in the areas of *diagnosis* (e.g. *needle biopsies*) and *treatment* (e.g. *radiation therapy*).

As this thesis is concerned, we will have a closer look at one of these minimally invasive procedures, the so-called *endovascular aneurysm repair (EVAR)* which is used for the treatment of *aneurysms*—more specifically, *abdominal aortic aneurysms (AAAs)*. The intervention has to be conducted under constant image-guidance in order to help the surgeon navigate through blood vessels and to insert a tubular device, the *stent graft*, which is supposed to seal off the blood flow from the damaged part of the aorta. The procedure is described in detail in section 4.2.

An *automatic detection* of the stent-graft *during* as well as *after* the intervention can assist the surgeon in various ways: *Intraoperatively* it could provide enhanced visualization useful for navigation. *Postoperatively*, i.e. for assessment of the stent graft placement and for monitoring purposes, it may be a time-saving tool by eliminating the need for manual segmentation which is commonly required in order to make measurements and to verify the correct pose or to control a possible *migration* in between of monitoring sessions. In addition, an automatic process is *objective* and *reproducible* in contrary to the manual one. These are only some examples of the benefits of an automatic stent graft detection.

This thesis tries to give an overview of what is required to accomplish this goal, analyzes the solution options and presents a new approach to address the problem. The outcome is a method based on *model-to-image* registration which fits a geometrically-founded deformable parametric model of the stent graft to the image of interest and, thus, segments the object *inherently* by detecting its pose and shape in the image. Even though there is still room for improvement concerning some details of the presented approach, experiments demonstrate that the method works in general and can serve as a basis for further research. Ideas on addressing open challenges and beyond are given at the end of the thesis.

2. Computer Scientific Background

Nowadays, computers are integrated in all our daily lives. Many processes—whether it may be production, banking, medical interventions, etc.—have become unthinkable without them. Thus, the research areas of computer science have spread into all these domains.

This thesis deals with one specific field, namely, *computer-aided medical procedures*. In this introductory chapter let us slowly approach the specific topic from the computer scientific side before proceeding to the technical and the medical part. The next section provides the necessary background on *digital images* and *image processing* in general, while the subsequent one introduces the field of *computer-aided medical procedures* and *medical imaging* as well as *image-guided interventions* in particular. An overview of relevant medical imaging modalities and devices is postponed to the next chapter (3).

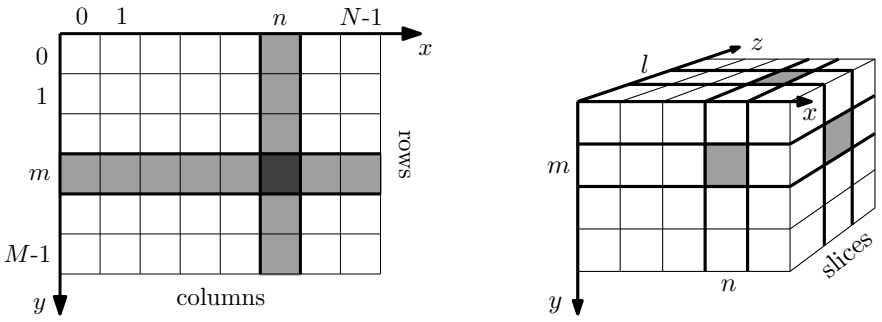
2.1. Digital Imaging

Throughout the thesis we are going to be concerned with *(digital) images* and *image processing*. Hence, it makes sense to clarify what we actually mean when talking about *digital imaging*—along with some common terms that accompany this subject.

2.1.1. Digital Image Representation

When we want to take pictures of the world we live in, and the objects that surround us their images have to materialize in some way that we can look at them. An *analog camera* does this by catching the light emitted or reflected from the objects in the field of view and projecting it through a lens onto a light-sensitive photographic film. Note that a similar approach has been used in the early days of *medical imaging* (without a lens, though): The person or any material to be examined has been placed between the X-ray source and a film optimized for sensing X-ray radiation.

Image Digitalization. Nowadays, however, the overwhelming amount of pictures—whether they may be taken by a *medical imaging device* or a *digital camera*—are stored on *digital media* (like solid-state disks or hard drives as used by computers) and visualized by *displays* or *monitors*. Since such digital media can only store *discrete* values (typically represented in binary numeric form by 0s and 1s) and has a *finite* capacity it is not able to represent the analog real-world information in full detail. Therefore, a lossy conversion of this information is necessary—i.e. a projection of the infinite-dimensional world space to a finite-dimensional image space—in order to retrieve a *digitalization* of



(a) 2D digital image. (b) 3D digital image.

Figure 2.1.: Digital image representation as a grid in 2D (a) and 3D (b).

it. This process is commonly called *digitizing* and involves *sampling*, or *discretization*, and *quantization*. We do not want to go into detail here because this is done in section 8.1.3 when we need to digitize the model we develop in the corresponding chapter. Just so far: The result of the digitalization is a *discrete and finite set* of *discrete values* that represents an image in digital form for storage on digital media or for *digital image processing*.

Digital Image Types and Components. There are basically two types of digital images: *Vector* images are described by geometrical primitives (such as points, lines, circles, curves, polygons, etc.) which are based on mathematical equations. (We are not going to be concerned with this image type.) *Raster* images, on the contrary, are composed of a finite set of entities, the *picture elements*. The latter can hold (one or several) digital values and are usually modeled as *matrix* or *grid* ordered in *rows* and *columns* (see figure 2.1(a)). A picture elements is referred to as *pixel*, or occasionally just *pel*[1]. Each pixel can either store exactly one value for *grayscale* images or consist of several components (e.g. red, green, and blue) for *color* images. These values represent the *brightness* of the given color at a specific point. For grayscale images a pixel carries only *intensity* information. In order to read the values from the digital storage the pixels can be addressed according to their position in the grid. Thus, the latter makes up a *coordinate system* with *axes* commonly denoted with x and y.

Image Dimension. The grid on which the pixels are laid out defines the image dimension of a raster image, i.e. it typically has a certain *width* and *height* making a total of two dimensions. However, we have to make clear that a digital image can also have three or more dimensions. 3D images consist of *volume elements* instead of picture elements and the analogue of a pixel is called *voxel* which, likewise, has a certain color or intensity defined by the values it contains. The third dimension is made up by stacking several 2D grids one after another. The additional layers are called *slices*

[1]The terms pixel and pel simply arose from abbreviations of the lengthy “picture element”.

and are addressed by an additional axis z. Figure 2.1(b) illustrates this. Using a fourth image dimension we could compose *motion images* (as opposed to *still images*). Medical imagining devices generally produce 2D, 3D, or 4D grayscale raster images. However, we will not consider 4D images in this thesis and, thus, details are spared here.

Terminological Conventions in this Thesis. The different types and the issues about dimensionality can make it confusing to talk or write about digital images. In order to clarify the usage of the terms in this thesis we obey the following conventions: For convenience, the term *image* implies that we are dealing with a *digital raster* image (as long as not stated otherwise). In our case, we always have *greyscale* images, i.e. one value per pixel or voxel defines its *intensity*. Keep in mind that when we use the term "image" nothing is stated about its dimension. Thus, it can refer to both a 2D or 3D image here. If we want to point out that only one of these two types is meant we may explicitly add the dimension—*2D* or *3D*—or use the term *volume* for the latter.

2.1.2. Digital Image Processing

Digital image processing is concerned with *analyzing* or *modifying* digital images by using computer *algorithms* in order to examine the contained features, to enhance the image quality, or to produce a desired visual effect. Typical tasks are *feature extraction*, *pattern recognition*, *segmentation*, *classification*, etc. *Preprocessing* operations (like *smoothing*, *morphological operations*, *edge detection* and other *filtering operations*) frequently precede the segmentation of certain objects and *postprocessing* (like *connected component analysis* or *boundary* and *surface extraction*). The thesis is not meant to discuss all the various image processing algorithms as there are a lot of textbooks out there for that purpose. At this point, we want to focus on the processing of *medical images* and the methods we need for our task—especially on segmentation as means to detect our object of interest, the stent graft. For a start, let us take a look at computer-aided medical procedures in general and, more specifically, introduce medical imaging and image-guided interventions.

2.2. Computer-Aided Medical Procedures

As stated in the introduction (chapter 1) computer-aided medical procedures play an important role in the clinical daily routine. The huge amount of image data collected throughout medical diagnosis and treatment needs proper management and processing. Thus, it is not surprising that it makes up a wide field of research these days.

At the *TUM* the *Chair for Computer Aided Medical Procedures & Augmented Reality (CAMPAR)* headed by Prof. Dr. Nassir Navab and Prof. Dr. Gudrun Klinker is concerned with the analysis of modern hospitals' and physicists' needs and tries to develop methods and solutions that enhance the complete medical workflow. This does not only include *image processing* of medical image data and *image-guided interventions*

(a) X-ray of hand. (b) Wilhelm Conrad Röntgen.

Figure 2.2.: (a) X-ray image of Anna Berthe Röntgen's hand. The image clearly shows the bones of the hand and rings on a finger. (b) Photograph of Wilhelm Conrad Röntgen.

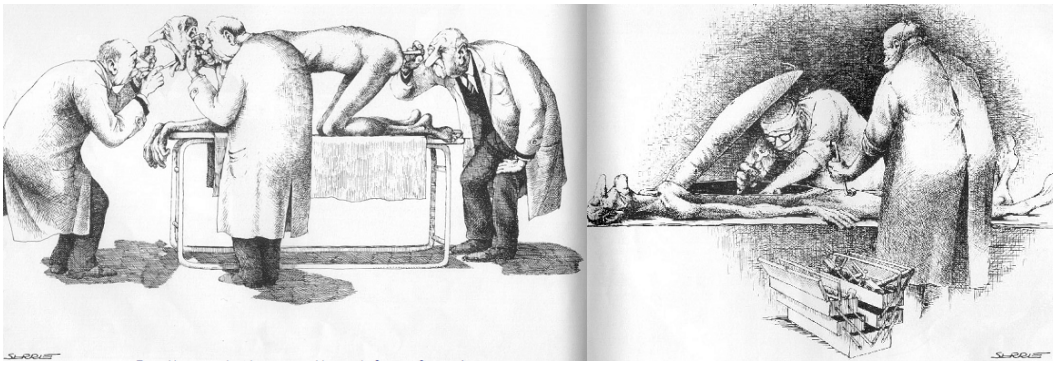
but spans the whole medical domain—reaching from advanced *visualization* over *medical information systems* and *databases* to the in-depth analysis and optimization of the *medical workflow*.

We will not detail all the individual subjects here as this would fill a whole book (e.g. [PC08] is solely dedicated to image-guided interventions) but, instead, focus on procedures that are relevant for the thesis. The latter are basically related to the medical image processing domain, especially *segmentation* and *registration*, as an essential part of *image-guided surgery*—but let us get it on slowly by first giving an overview of the history of medical imaging and the image data and imaging devices we are going to be confronted with.

2.2.1. Medical Imaging

Let us first review the history of medical imaging along with the state-of-the-art and an outlook on future developments.

History of Medical Imaging. The discovery of the capabilities of X-rays concerning medical applications by Wilhelm Conrad Röntgen (figure 2.2(b)) in 1895 in commonly referred to as the historical beginning of modern medical imaging. The first medical



(a) Observe the system output non-invasively. (b) Open up the body (open surgery).

Figure 2.3.: What physicians could do without images. (From "Humour noir et hommes en blanc" – de Serre – Cartonné.)

image published is the famous picture of his wife's hand which clearly shows the hand's bones—as well as the rings she was wearing (see figure 2.2(a)).

That moment dramatically changed the medical routine. Hospitals around the world started using X-ray devices in order to get a glimpse into the body of a patient *without* opening it up. Two famous humorous illustrations show the situation before the introduction of medical imaging devices (see figure 2.3). The latter induced a paradigm change of traditional (*invasive* or *open*) surgery to *minimally invasive* interventions making it possible to examine a patient's internal state using image-based diagnosis, treat diseases my means of *minimally invasive surgery*, and only opening up the body when really necessary.

It took a while until mankind found out about the dangers and risks of X-rays. While in former times, images were taken under constant radiation it is now the objective to keep the radiation dose as low as possible to safe both physicist and patient from damage. Improvements to the original devices as well as completely new ones based on different techniques helped getting closer and closer to this goal. In addition, computer-aided procedures play an important role by facilitating novel navigation methods which reduce the amount of images that have to be taken for the process.

State-of-the-Art of Medical Imaging. Meanwhile, medical imaging is at a stage where various different imaging devices are used at a daily basis in hospitals around the world. They have become an essential part of the clinical routine during all stages of the medical workflow like diagnosis, preoperative planning, intraoperative navigation and postoperative monitoring and treatment.

The list of medical imaging modalities in use is long: *Computed tomography (CT)*, *fluoroscopy*, *magnetic resonance imaging (MRI)*, *ultrasound (US)*, *positron emission tomography (PET)*, *single photon emission computed tomography (SPECT)*, etc. Ad-

ditionally, the imaging methods may be combined with contrast-enhancing techniques like using *contrast agent* to enhance the visibility of tissue such as vessels (as done for *angiography*). All the different techniques have their own strengths and application areas. It is up to the pertinent textbooks to give a detailed introduction to all the imaging modalities and devices. Chapter 3 gives an overview of the ones that are relevant for the thesis.

It is more than ever necessary that computer scientists develop methods that support physicians manage the vast amount of image data. Image segmentation, for example, is used to extract structures (like tumors, organs) and to create (3D) anatomical models which can be used in different stages of the medical workflow. Here are some examples:

Diagnosis: Make measurements on the dimensions to check for unnatural alterations.

Planning: Define tissue to be resected or plan paths through vessels or tissue.

Intervention: Overlay the model and/or plan with the body or interventional image data for *navigation*. Intraoperative segmentation can provide enhanced visualizations and support the navigation procedure. In conjunction with registration, preoperative 3D image data can be used intraoperatively to extend the (mostly 2D and low-quality) interventional images.

Monitoring: Make measurements on the dimensions to control the effect of a certain treatment.

Many interventions already make use of this technique and the list of applications is constantly growing. Concerning the field of image-guided interventions in particular, [PC08] presents a comprehensive snapshot of the state-of-the-art.

Future of Medical Imaging. As all the various imaging devices have their own distinctive qualities and application areas there is an ongoing progress of improvements to the individual devices as well as the *combination* of different imaging modalities to get the most out of all techniques. This can drastically increase their value for clinical applications. However, the so-called *image registration* is only one of the current fields of research.

Some methods have found their way into the clinical routine already while others are still in development. Key techniques include *computer-aided navigation*, *image-guided surgery*, *patient/instrument tracking*, *in-situ visualization* of medical image data (using *augmented reality*), *(semi-)automatic segmentation* etc.

Aside from that, the current situation in hospitals is still kind of inconvenient: A lot of different monitors and devices are placed more or less arbitrarily in the operating room making it very cumbersome for the surgeon to access the right information at the right time. However, there are efforts to improve the situation by the development and implementation of specially designed medical environments—like e.g. the *cardiac catherization lab (CATH-lab)*—that meet the requirements of specific interventions with

all the necessary devices at the proper place and the important information bundled and visualizable through one ore several interactive displays.

Further concepts and developments may change the way medical images are handled in conjunction with all other information from the medical workflow.

2.2.2. Image-Guided Interventions

Image-guided interventions are a major part of computer-aided medical procedures. The term is very general as it covers both *therapeutic* and *surgical* procedures conducted under image-guidance. These are gaining more and more importance, because they facilitate *non-invasive* or *minimally invasive* interventions. In opposition to their invasive counterparts, the patient's body does not have to be opened up for diagnosis or treatment. This means less blood loss and, hence, less stress for the patient. Here are just a few examples of applications that already have found their way into the clinical routine:

Diagnosis: Image-guided *needle biopsies* and similar minimally invasive procedures allow to take samples of tissue or body fluids for examination.

Surgery: Minimally invasive image-guided operations, e.g. in the areas of *orthopedic*, *cardio-*, and *endovascular surgery* provide alternatives to open surgery. Our focus, *endovascular aneurysm repair (EVAR)*, also belongs here.

Treatment: Mostly *radiation therapy* and similar non-invasive or minimally invasive procedures that rely on image-guidance allow for treatment without opening the body.

These interventions typically require computational support in order to assist the surgeon plan, carry out and evaluate the intervention. *Visualization* of image data as well as intraoperative *navigation* play an important role and are facilitated by techniques like *segmentation*, *registration* and *tracking*. A detailed introduction to the field of image-guided interventions presenting a snapshot of the state-of-the-art is given in [PC08].

Before focusing on our specific intervention and the methods we want to use in order to enhance it, let us take a look at the imaging modalities that are related to the procedure.

3. Imaging Modalities

As already stated, there is a great variety of imaging modalities in use: *Computed tomography (CT)*, *fluoroscopy*, *magnetic resonance imaging (MRI)*, *ultrasound (US)*, *positron emission tomography (PET)*, *single photon emission computed tomography (SPECT)*, etc. Introducing them all is beyond the scope of the thesis and, consequently, just the relevant ones are explained in the following.

We are basically going to be confronted with two different imaging modalities, namely, *fluoroscopy* and *computed tomography (CT)*, which differ in their individual imaging qualities as well as their application areas. Both of them can be enhanced by a technique called *angiography*.

3.1. Computed Tomography (CT)

CT images are produced by CT scanners[1] which utilize X-rays to form the images. The device consists of an X-ray *source*, or *generator*, and a *detector* which are mounted on a *gantry*. The latter can rotate around the patient who lies on a movable bed. As the gantry rotates and the bed moves several images of the patient's body are taken which can be mathematically reconstructed into a complete volume composed of slices through the patient. The result is a 3D image with high resolution and excellent quality. Consequently, the device is typically used for *diagnosis*.

Although the speed of image acquisition and reconstruction constantly rises it is still one of the limiting factors that frequently prevents the device from being used intraoperatively. Other issues include its huge dimensions and low mobility along with the limited work space around the scanner. The patient has to be inside the gantry for imaging where it can hardly be reached by the physician's hands that have to guide instruments—not to mention other personnel that usually assists the surgeon. This is the main reason why fluoroscopy is commonly preferred for image-guided interventions.

3.2. Fluoroscopy

Just like CT scanners, fluoroscopy systems are based on X-ray radiation. Both *generator* and *detector* are mounted on a curved rack called *C-arm* in a way that they are facing

[1] The invention of the CT scanner in 1972 goes back to the British engineer Godfrey Hounsfield working for EMI. While the company is commonly known for the recording and marketing of popular music, it also maintained a research department (EMI Laboratories). Interestingly, the development of CT was likely to be driven by the world-famous music group "The Beatles", whose success may have helped raising the urgently needed funds (according to [PC08]).

(a) CT scanner. (b) C-arm.

Figure 3.1.: Pictures of CT scanner (a) and C-arm (b).

each other. The C-arm is mobile and can be moved to and rotated around the patient's bed which makes it possible to acquire images from arbitrary angles. However, only 2D images are produced[2].

On the one hand, this is an advantage because the lack of a reconstruction process make the device suitable for real-time applications. On the other hand, 3D information is missing and overlapping structures are hard to distinguish in the 2D projections. Nonetheless, fluoroscopy systems are the major device used for image-guided interventions due to their superior mobility.

3.3. Angiography

Since both CT and fluoroscopy utilize X-rays they are not well-suited for the visualization of soft tissue. The problem with this is that the surgeon has to navigate through blood vessels during our kind of intervention which are almost transparent in X-ray images.

Magnetic resonance imaging (MRI) is actually more appropriate for imaging soft tissue. However, the method has other drawbacks mainly referred to the problematic behavior concerning ferromagnetic material. The strong magnetic field attracts it and heats it up and, consequently, items made of it have to be put out of the way during imaging. Furthermore, patients' implants and surgical devices can cause image distortions. In addition, high purchase costs limit the availability of MRI devices in

[2]Modern fluoroscopy systems allow for the generation of 3D images from a couple of 2D slices via *interventional 3D reconstruction*. They are of lower quality than CT scans, though, and during interventions typically 2D images are still used.

hospitals. On top of that, their dimensions raise the same issues as already mentioned for CT scanners. Due to these facts, MRI is not particularly suited for image-guided interventions.

There is another solution to our problem called *angiography*[3] which can be used to visualize blood vessels. A radio-opaque (i.e. X-ray absorbing) *contrast agent* is inserted into the blood vessel through guide wires and catheters. An image can then be taken using X-ray-based modalities and is called *angiogram*. While, strictly speaking, angiography refers to *projectional radiography* (such as fluoroscopy) it can be used in conjunction with CT where the method is commonly denoted *CT angiography (CTA)*.

[3]The term "angiography" stems from the greek words "angeion" (vessel) and "graphein" (to write or record).

4. Medical Background

In this chapter, relevant medical terms and procedures are introduced. This basically involves the definition of an aneurysm in the first section (4.1) along with related terms as well as general facts and treatment options. Another section (4.2) is dedicated to the specific intervention this thesis deals with: *Endovascular aneurysm repair (EVAR)*. The workflow of the procedure is described including details on *diagnosis* and *preoperative planning*, the actual *intervention*, and *postoperative follow-up examinations*. Furthermore, the challenges of the intervention for the surgeon are examined along with their meaning for the patient. The role of computer science is pointed out by proposing possible solutions to these issues.

4.1. Aneurysms

4.1.1. Related Medical Terms

Every human body contains an intricate network of strong and flexible blood vessels called *veins* and *arteries* whose purpose is to carry blood to and from the heart in order to nourish body cells. The *aorta* is the largest blood vessel in the body (about the diameter of a garden hose) and makes up the root artery which is attached directly to the heart and successively branches into smaller arteries. It carries oxygen-rich blood away from heart and distributes it to the arteries supplying the entire body. It runs from the heart through the chest (*thoracic* aorta) and belly (*abdominal* aorta) and finally splits off into the *iliac arteries* that serve the legs.

An *aneurysm* is a blood-filled bulge or ballooning of a blood vessel caused from blood pressure due to weak vessel walls. Whilst it can occur anywhere in the body the most common location is the abdominal aorta including the iliac arteries (*abdominal aortic aneurysm (AAA)*), or the thoracic aorta (*thoracic aortic aneurysm (TAA)*). In this thesis, we will focus on AAAs bearing in mind that the situation for TAAs is similar. Figure 4.1 shows the aorta and an AAA.

Although an aneurysm does not necessarily pose a health threat to affected people there is the risk of a *rupture*, i.e. the bulge may become weak and the force of normal blood pressure can cause its bursting. This is a very serious and critical medical incident accompanied by severe pain and massive internal bleeding (*hemorrhage*). Without *immediate* emergency surgery, a rupture causes the patient's death.

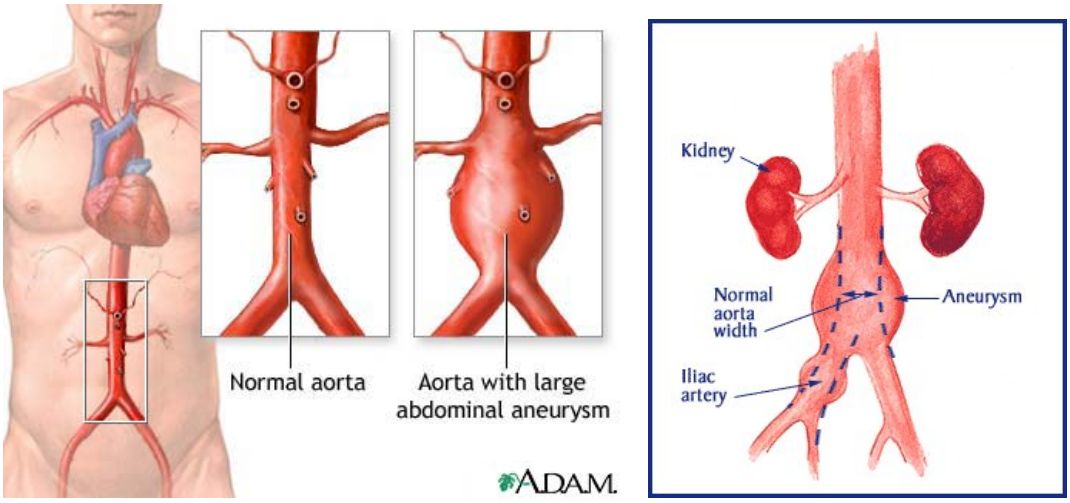


Figure 4.1.: Aorta with abdominal aortic aneurysm (AAA).

4.1.2. AAA Facts

There are about 1.5 million patients suffering from AAA in the U.S. with approximately 200,000 new cases each year. Over 15,000 people in the U.S. die every year from ruptured AAAs. The disease is the tenth leading cause of death in men over 50. An estimated 1 million men and women worldwide are living with undiagnosed AAAs.

4.1.3. Causes and Risk Factors

Aneurysms are caused by a weakness in the wall of the blood vessel. The reasons for such a weakness to occur are not fully understood yet. In some cases, the artery may have become inflamed, e.g. due to clogged arteries (*atherosclerosis*). However, it may also be related to heredity, injury, or other diseases. Known risk factors include the person's age (50+), cigarette and tobacco consumption, atherosclerosis, high blood pressure (*hypertension*), high cholesterol (*hypercholesteremia*) and genetic factors.

4.1.4. Symptoms and Diagnosis

Most people do not have any symptoms. In some cases, the following can be observed (esp. when an aneurysm expands): Chest or lower back pain (AAA/TAA); coughing, hoarseness or difficulty breathing (TAA); throbbing in the middle or lower part of the stomach (AAA).

Even though some aneurysms enlarge quickly they usually grow slowly and many never rupture. Thus, they often go unnoticed and in most cases are identified during routine medical exams or treatment of another disease where images of the chest and abdomen are taken using different techniques (including MRI, CT, fluoroscopy, US,

etc.). All of these are also used for diagnosis in case of an acute suspicion.

4.1.5. Treatment Options

There are basically three treatment options for an AAA or TAA depending on various factors like the shape and stadium of the aneurysm and the age and general health condition of the patient. They are summarized in the following paragraphs.

Watchful Waiting and Medical Treatment. Not all aneurysms are dangerous and need surgery. If the aneurysm is small (about 2 inches in case of an AAA or TAA), and does not grow rapidly a "wait-and-see" approach might be applicable, that is to say, a watchful waiting including regular monitoring via appropriate imaging techniques. It may be accompanied by medication (e.g. beta blockers that lower blood pressure) as well as changes in lifestyle (e.g. a balanced diet to lower blood cholesterol, moderate exercise, quitting cigarettes and tobacco use).

Open Surgical Repair. Treatment using open surgical repair involves opening the body with a large cut (*incision*) into the chest or abdomen. The damaged area is separated surgically from the vessel and replaced with a synthetic tube called *(aortic) graft*. It is quite a massive intervention that is performed under general anesthesia involving *cardiopulmonary bypass (CPB)* via a heart-lung machine, and takes about 3 to 4 hours. It usually requires the patient to stay for 3 days in an intensive care unit and remain in hospital for 7 to 10 days. Thus, recovery time is typically about a week but it can be as high as three months bearing risks due to anesthesia and the large incision (like infection, etc.). However, it is a good option for a long-term success as the damaged area is completely removed.

Endovascular Aneurysm Repair (EVAR). *Endovascular aneurysm repair (EVAR)*, is a less invasive alternative to open surgical repair using a special device called *(endovascular) stent graft* that gets placed inside the damaged area of the blood vessel to seal off the aneurysm from the normal blood flow from the inside and to reinforce the area of the vessel weakened by the aneurysm to prevent rupturing. The procedure is *minimally invasive*. Depending on the type of aneurysm and the used stent graft, it typically requires one or two little incisions under local anesthesia in the groin area through which *delivery catheters* are inserted in order to place the folded parts of the stent graft in position, where they get released (opened up) and fastened. Details on the intervention are explained in the following section (4.2). Being minimally invasive, recovery time is usually very short (typically 1 to 4 days) and patients can get back to normal activities within a few weeks. Nevertheless, periodic follow-up visits are inevitable to monitor the success of the treatment, and to detect complications like stent graft migration/fracture or limb kinking/occlusion.

EVAR vs. Open Surgery. In general, the minimally invasive EVAR procedure means less stress for the body. Open surgery, on the other hand, has known long-term success. The patient's age and general health condition may make him ineligible for open surgery, though. Every case is unique, and some may not be treatable using EVAR. For example, the aortic aneurysm may not have the right shape or may be too close to other organs. There are situations where either of the two treatment options may be chosen. Aside from these issues, table 4.1 compares these two procedures.

The long-term success rate of EVAR is still to be examined. [PVB+04], for example, takes a step in this direction. Concerning *ruptured* AAAs, recent studies showed that the *30-day mortality rate* is lower for patients treated with EVAR (about 22 %) as opposed to open surgery (38 %). Furthermore, the minimally invasive procedure has a lower rate for total systemic complications (28 % in contrast to 56 % for open surgery). [VvSH+07] compares 10 studies and provides a good starting point for further information.

4.2. Endovascular Aneurysm Repair (EVAR)

Endovascular stent grafting, or *endovascular aneurysm repair (EVAR)*, denotes the treatment of AAAs using *endovascular surgery*. In the following, let us focus on this intervention as exemplary (and most common and advanced) form of vessel repair. It is almost identical to the one used for TAAs and similar procedures exist for other kinds of vessel diseases. However, the latter ones might incorporate different types of stents and stent grafts (as explained in section 4.2.1).

EVAR is a sophisticated procedure requiring the joint knowledge and capabilities of vascular surgeons and radiologists. It is performed in a medical environment called *CATH-lab (cardiac catherization lab)* which allows physicians to navigate through the body with wires and catheters using imaging devices and monitors. It eliminated the need for some types of traditional open surgery.

The overall procedure includes diagnosis and preoperative treatment planning, the actual intervention and periodic follow-up examinations. This is illustrated by figure 4.3. First, we will take a look at *stent grafts* which are used for treatment before going into the details of the complete workflow.

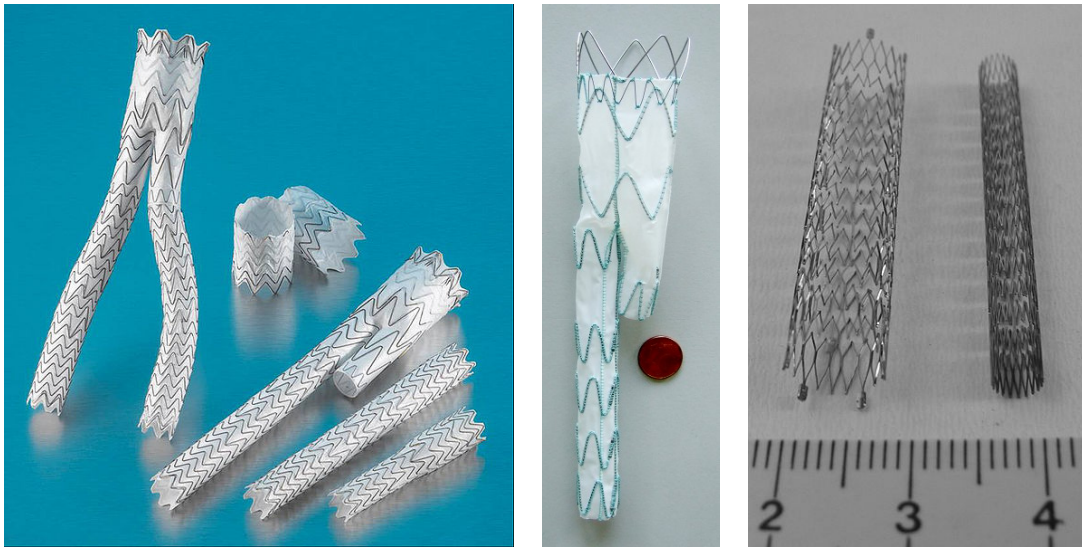
4.2.1. Stents and Stent Grafts

Stents and stent grafts exist in various shapes and configurations. Image 4.2 shows a collection. In general, stents are tubes which can be inserted into body conduits for the treatment of diseases commonly related to flow constrictions of body fluids, or to ease the access to affected passages for surgery. *Stents* are solely made from metal webs or meshes[1] while *(endovascular) stent grafts* are composed of a synthetic fabric tube, the

[1] The first stents in use were made from bare-metal. More recent ones—some of them still in development—are drug-eluting stents, stents with biocompatible surface coatings and absorbable stents (metal or polymer).

	Open Surgery	**EVAR**
General information	Traditional surgery and accepted standard (in use for over 15 years)	Established alternative (used over 10 years, use increasing)
Invasiveness	Major intervention involving open-chest/abdomen operation and *cardiopulmonary bypass (CPB)* via heart-lung machine (⇒ high blood loss and stress on body)	Minimally invasive involving 1–2 small incisions (⇒ less blood loss/trauma, reduced pain)
Anesthesia	General anesthesia required	Epidural (regional anesthesia) and general anesthesia possible (dependent on surgeon and patient's preferences)
Treatment effect	Diseased tissue is completely removed and replaced by an endograft	Aneurysm is sealed off from blood flow by a stent graft
Long-term success	High long-term success rate	Successful to date, long-term results not fully studied yet
Hospital stay	1 week on average (sometimes stay in intensive care unit necessary)	A few days on average (usually little or no stay in intensive care units)
Recovery time	Several months of recovery on average	A few weeks on average
Complications (interventional)	Risks due to general anesthesia and the large incision (high blood loss, infection, etc.); higher mortality rate than EVAR	Risk of aneurysm/aorta rupture slightly higher than for open surgery, but hardly any blood loss, and lower rate of infections and systemic complications
Complications (postoperative)	Rare since diseased tissue is replaced by graft; the graft is sewn in a way that it is rigidly fastened to both ends of the severed aorta	Stent graft migration, fracture, limb kinking/occlusion ⇒ blood leakages, blockage of the flow of blood (⇒ secondary intervention necessary)
Follow-up examinations	Hardly necessary	Periodic visits to monitor stent graft and to detect complications

Table 4.1.: EVAR vs. open surgery: Comparison of important issues.



(a) Stent graft components. (b) Trunk. (c) Stents for peripheral vessels.

Figure 4.2.: (a) Stent graft components (collection of main bodies, trunks and limbs). (b) Stent graft trunk component next to Euro cent. (c) Stents for peripheral vessels.

graft (e.g. from woven polyester), supported by a rigid structure, the *stent* (usually a metal web). *Endovascular grafts*, or *endografts*, lack the supportive structure and are used for open surgery to replace the diseased part of a blood vessel. Stent grafts, on the contrary, are used for EVAR and merely prevent blood from running into the aneurysm sack by blocking the blood flow from the aneurysm.

Stent Graft Components. Stent grafts used for EVAR (see figures 4.2(a) and 4.2(a)) usually consist of two or three parts: Either a *main body* and two *legs*, or a *trunk component* (including one of the legs already) and a *contra-lateral leg component*. The legs are supposed to sit in the iliac arteries, and are frequently referred to as *limbs*. Various sizes and configurations are available from different vendors, and, additionally to the commercial ones, custom-made stent grafts can be obtained. The components are folded up in order to fit into *delivery catheters*, and are capable of expanding to the pre-established diameter when placed and released in the artery.

Stent grafts help to bypass the area of the aorta weakened by the aneurysm to keep it from rupturing (bursting), and can be placed inside of the aorta without surgically opening the blood vessel or removing any tissue (as explained in the following section 4.2.3).

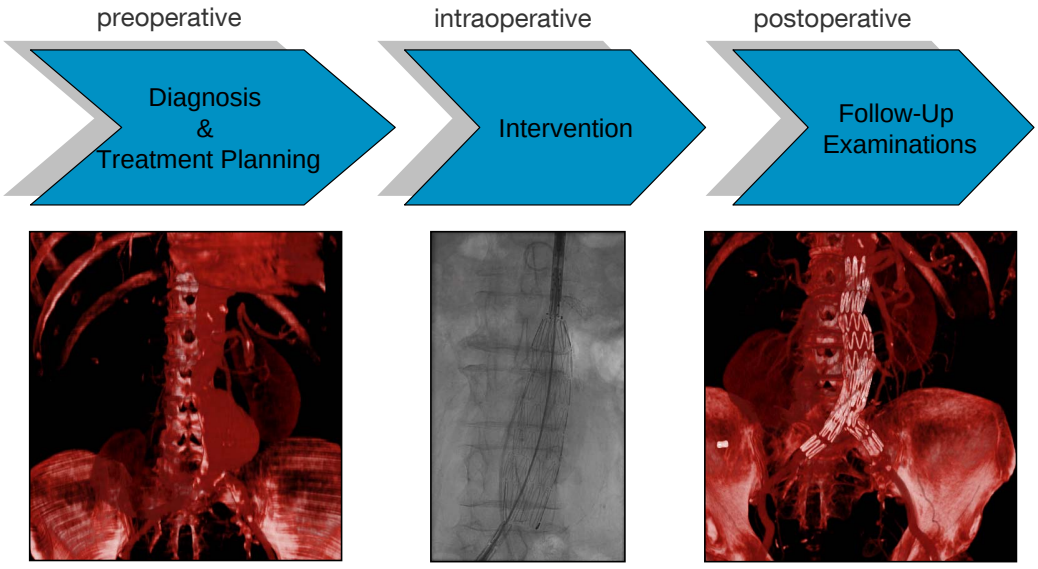


Figure 4.3.: The EVAR workflow: The preoperative part (left) includes diagnosis and treatment planning using CT volumes in order to examine and measure the aneurysm and its surrounding. The actual intervention (middle), i.e. the stent graft placement, is guided by interventional 2D images (fluoroscopy or angiography). Postoperative control and periodic follow-up examinations (right) require further CT scans to check the position of the stent graft which could migrate.

Terminological Conventions in this Thesis. Throughout this thesis we only consider *stent grafts* as used for EVAR, and no stents of any kind. Hence, when the term *stent* is used, it refers to the stent-part of a stent graft (as opposed to the *graft*).

4.2.2. Diagnosis and Preoperative Planning

Various imaging devices can be used for diagnosis. However, for treatment planning usually CT angiography is used. Computer-based vessel and aneurysm segmentation allows for making measurements on the aneurysm size and the surrounding anatomy, whereon the choice of the required stent graft is based. Furthermore, it facilitates path planning which is utilized for intraoperative navigation.

4.2.3. Intervention (Stent Graft Placement)

The patient is given either a full *general anesthesia* or an *epidural*, a kind of *regional anesthesia*. The choice is often up to the patient but may be affected by the surgeon's preferences or other circumstances. The actual intervention—i.e. the stent graft

placement—is carried out following these steps:

1. A small incision is made in the leg near the groin area and a catheter is inserted into an artery.
2. A delivery catheter is inserted through the vessel into the aneurysm to guide the stent into place. Advanced imaging methods support the surgeon in guiding the catheters to the area of the abdominal aortic aneurysm.
3. The stent graft is released from the delivery catheter whereupon it opens and expands to the pre-established diameter by itself under internal forces of the supportive metal web (see figure 4.4(a)).
4. The steps above may be repeated on the other leg in order to place another limb of the stent graft (which gets attached to the trunk component or main body).
5. Once the stent graft is in the correct position, the surgeon fastens it into place and removes the delivery catheter after verifying that no leakage occurs (i.e. the blood flow to the aneurysm is blocked).
6. The incisions are sutured.

Note that the entire procedure (well, except the parts concerning the incisions, I admit) is guided by modern imaging techniques—and this is exactly the point where *computer-aided medical procedures* developed by *computer scientists* come in. Section 4.2.5 bridges the gap between computer science and medicine by examining the current situation, as well as challenges for physicians carrying out EVAR interventions—along with its meaning for the patient, and suggestions on improvements using computer-aided methods.

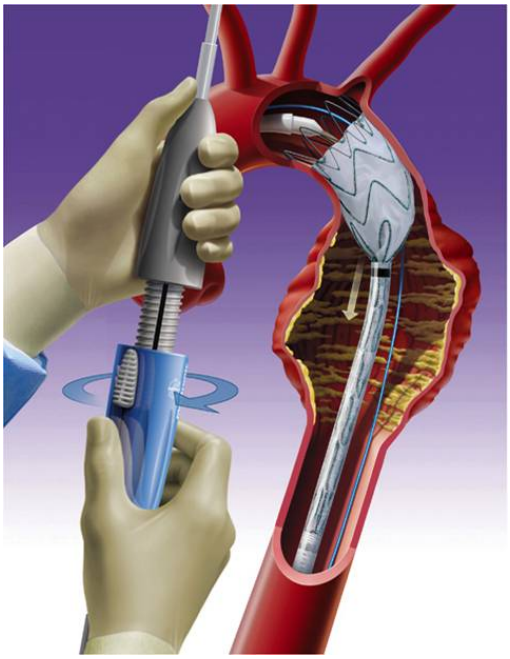
4.2.4. Postoperative Follow-Up Examinations

Periodic follow-up examinations are required in order to detect complications at an early stage. The latter include blocking of the blood flow and leakage into the aneurysm sac, induced by stent graft migration or fracture, or limb kinking or occlusion. In these cases, a secondary intervention may be necessary.

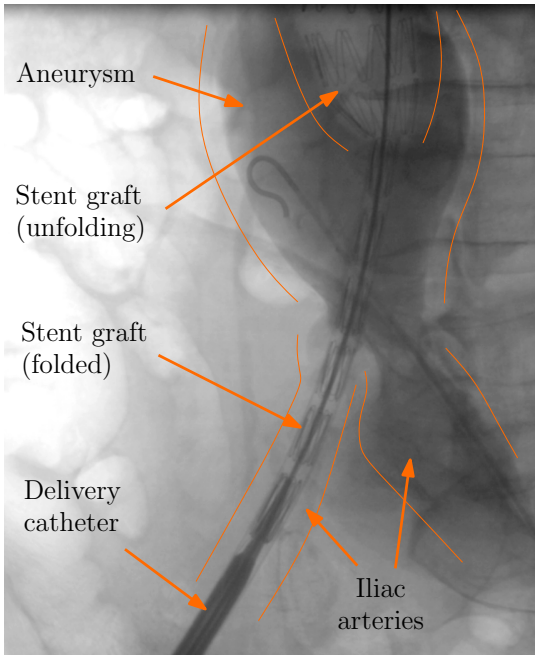
Monitoring of the treatment success usually involves CT(A) scans, and requires time-consuming manual segmentation of the stent graft in order to make measurements on migration, torsion, or other deformations. Again, computer-aided methods may yield improvements by automation of the process.

4.2.5. Challenges of the Procedure and Role of Computer Science

Surgeons have to face a couple of challenges both during the intervention and concerning postoperative monitoring. Let us summarize the current procedure and examine these challenges, their meaning for the patient, and the role of computer-aided medical procedures.



(a) EVAR: Release of the stent graft.



(b) Fluoroscopy of aorta with instruments.

Figure 4.4.: EVAR intervention: (a) The stent graft is released from the delivery catheter. (b) The angiography on the right hand side is what surgeons typically use for navigation of the instruments. Note that usually an image series is required to capture the moment when the contrast agent reaches the aneurysm.

Summary of the Current Procedure. The current procedure begins with diagnosis and preoperative planning usually involving CT(A) images (3D) whereas the complex intervention is carried out using angiograms or fluoroscopic images (2D). The surgeon uses many projectional images for navigation and stent graft placement. Postoperative control is done by manual segmentation of the stent graft which is required for measurements of its position and other properties in order to detect migration or deformation.

Challenges for the Surgeon and Threats for the Patient. The operation is complex and cumbersome for the surgeon for various reasons: He has to deal with various imaging devices and monitors as well as missing 3D information during the intervention. Postoperative control and monitoring requires time-consuming manual segmentation and assessment. However, the results of manual processing are neither objective nor reproducible. For the patient this means that he is completely committed to the skills, knowledge and experience of the physician as there is no objective verification or evaluation for both the success of the surgery and possible stent graft migration

Current Clinical Situation	**Challenges for the Surgeon**	**Threats for the Patient**	**Possible Computer-Aided Solutions**
Complex intervention due to missing 3D information during stent placement	Deal with several 2D images (angiography/fluoroscopy) taken from different positions	High radiation and contrast agent dose	Modern computer-aided navigation methods
Cumbersome postoperative control and monitoring due to intensive human interaction	Time-consuming manual segmentation and subjective assessment	Health threats due to lack of qualitatively meaningful validation	Computer-based objective verification of stent graft position and detection of migration

Table 4.2.: EVAR: Problems and challenges of the current procedure.

or other complications. In the worst case, misjudgments might impose a health threat on the patient.

Proposed Solutions and Role of Computer Science. Computer-aided medical procedures can provide solutions to these challenges. Modern image-guided interventions implement novel computer-aided navigation methods that incorporate 3D information from preoperative CT scans into the actual intervention by means of registration. The latter brings the given volumes into alignment with the interventional 2D image data, allowing for advanced visualizations. The manual segmentation process may be replaced by an automatic one which is time-saving, objective and reproducible. Table 4.2 summarizes these issues.

After the development of modern imaging devices "inseparable pairs" of physicians and physicists were formed in almost all hospitals between 1970 and 1990. Concerning the multitude of new systems and large sets of heterogeneous data, it is time to think about "inseparable triplets" of physicians, physicists, and computer scientists. (Nassir Navab, CURAC, 2004.)

Part II.

Problem Statement

5. Project Description

After the general introduction to our field of research providing the necessary computer scientific, technical and medical background it is now time to clarify the objective and motivation of this thesis along with the related work and the contribution to the community that is concerned with similar topics. At the end of the chapter, the required tasks to realize the project are summarized before the problem is analyzed in detail in the subsequent one. There, possible solution options and an overview of the developed approach are presented, too.

5.1. Objective

Let us first clarify what we have given: We have some kind of image which shows different anatomical structures as well as man-made ones. The image can either be 2D or 3D. In the 2D case it is typically a *fluoroscopic* image, frequently enhanced by *angiography* providing an *angiogram*. In the 3D case we commonly get a *CT volume* where, sometimes, contrast agent may also be used to produce a *CT angiogram (CTA)*.

Now, our general objective is to detect the stent graft which is visible in the images more or less *automatically*, i.e. without any user interaction, and to determine the stent's *position* (or *pose*) and *shape* in the image. Being aware of the difficulties of developing a fully automatic method—which is rarely achieved for similar problems—we can restrict the approach to be at least *semi-automatic*, for a start, with minimal user input for initialization. Some ideas on full automatization are given in section 10.2.5, though.

A little bit more specifically, the objective of the thesis is to examine and evaluate the use of model-based registration techniques as means to detect and segment the stent graft. The motivation for this approach is postponed to chapter 6 where some general thoughts on solving this kind of problems along with the analysis of solution options account for the choice. The reasons why detection of the stent graft is necessary at all are presented in the following section.

5.2. Motivation

The gaining need for and influence of image segmentation in all stages of the medical workflow (pre-, intra-, and postoperative) has already been outlined in section 4.2. Thus, let us just focus on the benefits of (semi-)automatic detection of stent grafts in (interventional) images, specifically.

EVAR is already a well-established medical intervention. However, there is still room for improvement using computer-aided medical procedures. In section 4.2.5 we already had a look at the challenges of the current EVAR procedure. Referring to table 4.2 we can make out two stages where information on the stent graft is essential: During the actual *intervention* and for postoperative *monitoring*.

Benefits for the Intervention. Typically, angiography is used for the current procedure to help the surgeon navigate to the aneurysm and place the stent graft at the correct position. The gathered 2D images can be hard to interpret due to overlapping structures and missing 3D information. Several images at different angles are required for the navigation process and to verify proper placement. This means that the patient is exposed to a high radiation and contrast agent dose.

The objective of modern minimally invasive interventions is to reduce the amount of X-ray-based imaging by implementing novel computer-based navigation methods. Registration allows for an overlay of interventional 2D images with preoperative CT volumes which extends the visualizable information during the procedure by a third dimension. The surgeon can rotate the volume interactively and look at it from different angles without the need for different projective radiographic images.

At this point, a detection of the stent graft in the interventional image can be very valuable. By knowing its pose and shape we are able to augment the registered CT volume by a 3D model of the stent graft, such that the surgeon can look at it from different sides and assert the placement or make out necessary corrections—with just one single interventional image.

Benefits for Monitoring. It has already been stated that after the EVAR procedure periodic clinical visits by the patient are required for monitoring purposes. This is due to fact that the stent graft can—and indeed usually does—move or *migrate* (frequently accompanied by kinking or torsion of the device). Blood flow and pressure is the main contributor to this movement. In order to prevent or detect leakage caused by migration the position of the stent graft has to be investigated on a regular basis.

Currently, this is done by visual inspection of the images and might include some manual segmentation of the stent graft and measurements concerning position, diameter, curvature, torsion and other deformations. This manual process, however, is neither objective nor reproducible. In the worst case, a misjudgment can pose a health threat to the patient. This fact constitutes for a qualitatively meaningful objective approach for the verification of the stent graft's pose and the detection of movements.

In order to accomplish this goal an automatic segmentation of the stent graft is the essential basis for the development of further computer-based methods that are able to classify different types and amounts of migration and to objectively examine the risks for patients.

For example, Mattes *et al.* [MSN+06] use nonrigid feature-based registration of stent grafts for the investigation of migration. To date, stent graft segmentation is accom-

plished in a manual fashion via thresholding or region growing. Hence, they may profit from an automatic approach, too.

Other Benefits. At the time of writing, engineers of the TUM are about to build up an experimental setup that is supposed facilitate the examination of the impact of blood flow on stent graft migration—aiming at fluid simulation that can predict such movements. An automatic stent graft detection could be very valuable for them in a way that they can use an objective and reproducible segmentation of the medical device which is the basis for a qualitatively meaningful study of the relevant physical relations (just like for the patient monitoring procedure).

That way, this work can be the foundation of various kinds of applications by eliminating the need for cumbersome, time-consuming and subjective manual segmentation of stent grafts in 2D and 3D images.

5.3. Related Work

The initial research for related publications yielded the fact that there is not much work out there addressing the problem of detecting stent grafts in 2D and 3D image data.

One of the first related works (known to the author) is the segmentation of aortic endografts by de Bruijne *et al.* as published in [dBNMV03]. However, these endografts are somewhat different from our stent grafts (see section 4.2.1). As the name implies, they are lacking the supportive metal stent since they are used for open surgery and are not required to unfold upon release from a delivery catheter. This means that the distinctive object features given by our segments are missing, too, in the used CTA images. In their work, this absence is compensated by the use of radiopaque markers attached to the outside of the device which enable the localization of the endograft and its central axis. Further gradient-based processing on the contours in a limited circular region around the axis yields the final segmentation. The overall approach cannot be applied to our problem because we are lacking markers and proper visibility of the graft in our images.

In [KRO+08] CT scans with ECG-gating are examined which provide insight in the motion of the stent graft caused by hemodynamic forces at different phases of the cardiac cycle. Automatic stent graft detection is done by segmentation on individual slices perpendicular to the centerline. Bright circular regions are visible at locations where the metallic stent penetrates the slice. Point detection methods and clustering algorithms are used to detect these regions. The position of the stent's centerline is estimated by fitting a circle through these points. The drawback is that the method is only applicable to 3D images. Furthermore, it is not a model-based approach. However, as we will see later, we would like to develop and use a model of the stent graft for several reasons. For example, various applications may require such a model.

Larrabide *et al.* [LRF08] used constrained simplex deformable models in order to developed a model of *intracranial stents* which is based on geometrical and physical

properties of the stent. They require it for a *virtual stenting application* which allows for releasing stents in arbitrarily shaped vessels and aneurysms. It is not used for stent detection in image data, though. Merely the idea of developing a geometrically and physically based model is noteworthy, because this is exactly what we are going to focus on.

In [ČAW+06] the individual vertebrae of the spine are segmented and registered using a two-stage approach. As the overall strategy of piecewise registration is similar to the approach presented here, their method may be related. The detection of the individual vertebrae is accomplished by a rigid registration followed by the application of a global deformation field which preserves the rigid nature of the vertebrae and elastically deforms the tissue around it. There are some fundamental differences between their approach and the one presented here: As already stated, we use a geometrically founded *deformable parametric* model, and, hence also require *nonrigid* registration. Furthermore, we do not apply a global deformation field after the registration in order to adjust the model locations. However, their method also incorporates prior knowledge of the spacial relationships between the objects, gathered from a slice-wise segmentation similar to [KRO+08]. Again, this approach is restricted to 3D images.

5.4. Contribution

Given the lack of a (semi-)automatic method for the detection of stent grafts in both 2D and 3D images, this thesis provides a first step towards addressing the problem. An approach based on segmentation via registration is developed and evaluated and the choice for this method is adequately reasoned. Experiments demonstrate that the implemented technique generally works (given reasonable prerequisites) and, furthermore, provide insight to open challenges. Ideas on addressing these issues as well as suggestions on future work complete the thesis.

Concerning the *computer scientific community*, on the one hand, it may be a valuable source for further research to anyone working in this field of computer-aided medical procedures. The *medical community*, on the other hand, may profit in the future by being provided with another computational tool that has the potential of making things easier in the clinical routine. Last but not least (I am sorry for running out of hands), affected *patients*—possibly including you and me—may benefit from being opposed to less radiation and contrast agent if the tool finds its way into the EVAR procedure.

5.5. Tasks

Now that we have a goal which is reasonably motivated let us start to accomplish it by summarizing the required tasks. We need to:

1. Analyze the problem (including a priori knowledge, prerequisites and requirements),

2. Figure out and reflect on existing methods and solution options,
3. Develop a new approach (if required),
4. Implement the approach (including the development of necessary methods and models),
5. Implement an application to test the approach,
6. Evaluate the approach using phantom and real patient data.

So, let us get it on with the first point.

6. Problem Analysis and Solution Options

Before explaining the theory and introducing the used methods and models this chapter is supposed to give insight to the general approach we want to take for our task. The latter is—as the thesis title states and as already explained in chapter 5—the detection of stent grafts in 2D and 3D image data. There are various different approaches and image segmentation methods that could be used for this purpose. Hence, at this point we like introduce and compare them, as well as choose one option as basis for our approach and motivate our choice. First, however, let us start with some general thoughts on addressing the problem and analyze the a priori knowledge we may be able to utilize.

6.1. General Thoughts on Addressing the Problem

Let us clarify once again what kind of problem we are confronted with: We have an image (2D or 3D) where different kinds of objects are visible. The latter include human bones (usually ribs and vertebrae) and soft tissue (e.g. blood vessels) as well as and man-made structures (like catheters and wires). However, we are only interested in one of these objects, namely, the stent graft in this case.

The image quality and, hence, the visibility of the structures in question is up to diverse image properties. The latter are partly affected by specific settings of the imaging device that are dependent on the desired output in conjunction with the expertise of the person using the device. As a consequence, the visibility of the object of interest might be disturbed by several factors like noise, artifacts, poor contrast, distortion, overlapping structures, etc.

Thus, a couple of general questions arise:

1. What do we know about the image and the object of interest including its appearance in the image?
2. What makes (the image of) the object of interest *unique* and, hence, distinguishable from other objects?
3. How is the object related to other objects and structures in the image?
4. What features are *still present*—and how *strong* are these—assuming the common disturbing factors?

5. And finally: How can we exploit our knowledge of distinctive image and object features?

In the following section we try to give answers to these questions—first by some general considerations on object features and appearance and then concerning our specific object, the stent graft.

6.2. Analysis of A Priori Knowledge

The analysis of a priori knowledge is an important step preceding the decision on—or development of—an approach to solve the problem at hand. It concerns the properties of the given image as well as the features of the object of interest along with their appearance in the image.

Common Image Properties and Object Features. This paragraph is merely supposed to provide a starting point for further considerations. A quick brainstorming could give us the following incomplete list of the given image properties and object features as well as their relation:

Image: Type (CT, MRI, etc.), dimension, size, resolution, ...

Object: Size, shape, material, ...

Object appearance in the image: Brightness, homogeneity, location/orientation (in the image or with respect to other objects), ...

Of course, it is the specific image properties and object features that define the object's appearance. With these general considerations in mind it is possible to approach our medical device.

Stent Graft Features. With regard to the upper paragraph, let us focus on the distinctive stent graft features. The following list is supposed to do this more or less completely:

Material: Metal (stent), synthetic fabric (graft).

Shape: Tubular; may be slightly bent; sometimes limb components; made of individual segments (thin, tubular and sine-like).

Location/orientation: Lies within the aorta; usually perpendicular to spine (2D: in front of it); segments stacked up on top of each other.

Appearance in the image: Graft hardly visible; stent bright (CT/CTA) or dark (fluoroscopy/angiogram), sometimes metal artifacts; individual segments visible as 2D/3D sine-like curves (contours overlapping in 2D).

As the graft is nearly transparent in the images of interest we have to focus on the visible part of the stent graft, namely, the *stent* which is made of metal wire. It is composed of individual segments forming a tubular sine-like shape. This design is motivated by the fact that the stent graft has to be compressed to fit into the delivery catheter and is supposed to unfold on release.

The nature of these curves, i.e. their shape and contrast in the given images is the most distinctive feature and, hence, is the essential basis for the detection method to be implemented. In addition, their positional relations both to each other and to other structures may also play an important role.

Factors Affecting the Features. Referring to the section 6.1 several factors affect the visibility of the features. Let us summarize them once again and examine their effect on the stent graft features as sketched above:

Contrast: Depending on the settings of the imaging device and surrounding structures the stent segments may suffer from poor and varying contrast.

Noise: Mostly Gaussian noise spread all over the image may decrease the visibility of the segments.

Artifacts: Metal artifacts may lead to frayed contours of the segments.

Distortion: In case of fluoroscopy/angiography the produced 2D images undergo a projective distortion which naturally affects the segments, too.

Overlaps: Also important for 2D images; other structures (typically catheter and bones) frequently overlap with the segments and cross the contours making them harder to identify.

Summary. Wrapped up, we have to find a method that is robust in respect to the factors analyzed in the previous paragraph. This means that the algorithm has to be able to cope with projective distortions, missing information and broken contours (due to noise or poor contrast) as well as overlapping structures. This also means that it is probably very helpful if the approach allows for the incorporation of our a priori knowledge of the object appearance (i.e. basically the stent segment shape) which we were able to identify above.

6.3. Review of Image Segmentation Methods

There is a great variety of *segmentation methods* that could be used for our detection problem. Thus, we want to just briefly outline some of the common techniques for the analysis of (not necessarily medical) images without going too much into details. A more comprehensive introduction to the methods examined here is given in [PXP00] and [PC08] (as referred to the medical context), and further image processing methods can

be found in [Jäh05] and other pertinent textbooks. Nevertheless, this section introduces and compares the different methods that may be suitable for the task at hand, and is intended to show their drawbacks—related to our specific application, of course—as well as to motivate the approach presented here.

6.3.1. Intensity-Based Segmentation

Intensity-based segmentation methods are very simple techniques considering only pixel or voxel intensity values in order to segment object. The most popular ones are *thresholding* and *region growing*.

Thresholding. This simple technique is well-known and, hence, does not require much introduction: It basically separates the image into two disjoint binary regions, *background* and *foreground*, by using predefined *upper* and *lower thresholds* on the pixel or voxel *intensities*. The major drawback is that it acts purely on the image intensity values. Two issues arise: First of all, it is necessary to define the thresholds. Although this might be done automatically in some cases by analyzing the image histogram it usually does not yield good results on medical images due to huge intensity changes (even in an intra-modal sense—not to mention inter-modality). Second, other structures with the same intensity as the object of interest are segmented, too. Third, the object may suffer from poor contrast and considerable intensity variations. These *inhomogeneities* usually makes a *manual adjustment* of the used thresholds inevitable, but even with a lot of twiddling the results are often *over-* or *under-segmented*. Furthermore, only points with same intensities are added to the segmentation, and not actual *connected components* that define a specific object. Thus, further processing by means of *connected component analysis* is required. The most serious drawback is that we are not able to incorporate any information on the *object shape*.

Region Growing. This method is similar to thresholding and acts on *intensity values*, too. Starting from a *seed point* pixels or voxels are added to the selected region based on their intensity difference to the seed. It gives a finer control over the regions added to the segmentation than thresholding, and has the advantage of resulting in *connected components* of *homogeneous regions*. However, it suffers from similar drawbacks: It may require a lot of *manual interaction* to get seed points and thresholds right in the case of intensity variations of the object. In addition, *overlaps* with other structures of similar intensity might cause the region growing algorithm run into them and segment them, too, resulting in *over-segmentation*. One example of where this could happen is the stent graft which might overlap with a catheter in interventional 2D images—the latter, however, is not supposed to be segmented. Again, we cannot make use of any prior knowledge of the actual object shape.

Figure 6.1 compares the results of thresholding and region growing.

(a) Original MRI of the head. (b) Thresholded image. (c) Region growing result.

Figure 6.1.: Comparison of thresholding and region growing: (a) The original MRI image of the head. The seed point is only used for region growing. (b) Thresholding segments pixels with same intensity values resulting in a binary image which contains a lot of structures. (c) Region growing, on the other hand, yields a connected component of a homogeneous region (i.e. the brain, in this case) starting from a seed point.

Further Methods. There are a couple of further intensity-based (as well as gradient-based) image segmentation methods, such as *split-and-merge*, *watersheds*, *live wire*, etc. However, details are spared here, because they all have in common that they lack the capabilities to make use of a priori knowledge of the object shape.

6.3.2. Higher-Level Segmentation

Higher-level segmentation methods do not only rely on image intensities but may also incorporate object features derived from preprocessing. Note that it is hard to draw sharp lines between these methods and intensity-based ones or the subsequent model-based approaches. Again, the following list is not complete, but the most common methods in the medical domain include *classification*, *snakes*, and *level sets*.

Classification. This kind of pattern recognition technique aims at partitioning the image into *classes* based on *intensities* or image *features*. There are two variants: *Supervised* classification makes use of a *training set* derived from priorly labeled image data. In a *training stage* the desired features are learned by the algorithm, which makes it possible to incorporate known information on the object features. Its drawback is the *manual labeling* process which, first, is *time-consuming* and, second, is *error-prone* because a representative training set has to be chosen. The latter greatly influences the segmentation quality and has to be specifically designed not only for the object of interest (including possible deformations) but also for the used *imaging modality*. This fact makes the choice of the training set and the labeling process a tedious work.

Unsupervised classification, on the contrary, does not make use of a training set but

tries to *cluster* the pixels or voxels automatically in a usually predefined number of classes by *minimizing intraclass distances* and *maximizing interclass distances*. The method, however, does not allow for the incorporation of prior knowledge and, hence, is *less robust* in general.

Snakes. These represent the classical means to use *parametric deformable models*. They have been originally designed for the 2D case where they represent *deformable contours*. Meanwhile, the approach has been extended to *deformable 3D surfaces*. The deformation is driven by *internal* and *external forces* making the snake reach a state of *minimal energy*. According to [PC08] a proper initialization of the snake is still an issue, because it may get attracted by structures surrounding the object of interest. Although there are efforts to incorporate shape information by learning from training sets they generally are not well-suited for making use of prior knowledge.

Level Set Segmentation. In contrary to the snake-approach (where an explicit parametrization is used) the evolving contour is defined as the *zero level set* of a higher dimensional function. The details are spared here, but we want to just point out the advantages: The level set can change topology—i.e. it can split or merge—with the level function remaining well defined. In addition, they are easier to apply on a discrete grid and in arbitrary dimensions. However, the implicit representation complicates the incorporation of prior knowledge and we will see that explicit shape models are more appropriate for that purpose.

6.3.3. Model-Based Segmentation

We have seen that most of the intensity-based and higher-level segmentation methods do not allow to make use of prior information on the object shape and appearance which we identified in section 6.2—or at least make it a tedious work. More promising techniques fall into the group of *model-based image segmentation*. Let us examine the most popular model-based segmentation methods which more and more find their way into medical image analysis.

Statistical Shape Models. Methods based on *statistical shape models* are gaining influence in the medical image segmentation community, because encoding prior statistical information on shape improves the segmentation results greatly in terms of accuracy and robustness to noise, artifacts and missing data which are common issues we are confronted with in medical imaging. Many different types and variants of statistical shape models exist, so let us just briefly introduce the most popular ones:

Active shape models (ASMs) represent shapes by a set of *landmarks*. Shape statistics are modeled by labeling training shapes with corresponding landmarks. Using these training shapes it is possible to compute a *mean shape* and *shape variation* (gathered from the covariance matrix) constrained to certain limits (dependent

on the training shapes). The segmentation process now reduces to finding the *pose* (typically a *rigid transformation* derived via registration) and the *shape* of the model (derived in an iterative manner) such that it matches the desired object in the image. An in-depth investigation of ASMs and their extension to AAMs can be found in [CT04].

Active appearance models (AAMs) extend ASMs by not only modeling statistics on shape but also on *appearance*. This is done by warping the training images to the mean shape in order to transfer them to the *shape normalized space* where texture information is finally extracted. The model is fitted to the image in a way that differences in both the shape and appearance component are minimized. This is a great idea which is useful for many applications. Unfortunately, in our case we are missing distinctive texture information.

Statistical deformation models (SDMs) are a further extension to ASMs. It originates from *atlas-based segmentation* whose objective is to solve the segmentation problem via *registration*. For this purpose a labeled dataset, the so-called *atlas*, could be used. The idea behind SDMs is not just to use a single atlas, but, similarly to ASMs and AAMs, the labeled training data serves as input for the learning stage. The key difference to the other methods is that a statistical model of the *nonrigid transformations* is built up that are able to map the images to one another (either within the training set or to new ones) and the object of interest is segmented via *nonrigid registration*.

All these methods have in common that they are very well-suited for the incorporation of prior knowledge on shape and/or appearance. The process of how this is done, however, is very cumbersome and time-consuming and the major drawbacks very much resemble the ones already mentioned for classification, i.e. the choice of a *representative training set* that is able to capture the desired range of variation along with the need for *tedious manual labeling*.

Nonetheless, SDMs come very close the approach we would like to pursue. The important idea is to have a model which we can register to the image whereby we get the segmentation *inherently*. However, there is an important difference to our method: We may skip building an atlas or a deformation model by incorporating the possible deformations into a *parametric model* based on the object's real *geometry* and *physical behavior*. Details on the model are presented in section 8.1. However, let us first outline the general approach.

6.4. Outline of the Developed Approach

In the last section, we compared different methods that could be useful for our task and already gave an idea where we are heading to. Let us just summarize the general idea once again in more detail before we come to the mathematical formalizations in the following chapters.

6.4.1. General Idea

As the last paragraph on SDMs stated, it is possible to detect an object of interest, i.e. the stent graft in our case, by using a *model* of it and performing a *nonrigid registration* which fits the model to the image. This yields the object's *pose* and *shape* in the image and, hence, a *segmentation* of it.

The approach used in the thesis is based on this idea. However, in contrast to SDMs or atlas-based segmentation we neither use an atlas nor a statistical deformation field, as we are in the lucky situation that we are able to develop a model of the stent graft that is based on its real *geometry* and *physical behavior* without too much difficulty. The idea is to treat the individual stent segments separately, because they can easily be modeled parametrically as tubular sine-shaped curve with a certain radius and amplitude. These segments can then be stacked up to form the complete stent while registration takes place on the individual segments one after another.

This approach resembles the so-called *piecewise rigid registration* as used in [ČAW+06] for the segmentation of vertebrae, but we have two major differences: First, we perform a *nonrigid* registration and, second, we use the fact that we know the approximate distance between segments in order to skip the manual initialization of *every* segment. That way, only the first segment needs a proper initialization which, at the moment, is still given by user input. In addition, the user specifies some information in advance, i.e. the approximate segments' dimensions and distances[1]. Note that this has to be done only once for each stent graft prior to the intervention or the monitoring routine and, hence, is not up to hard time constraints. Please, also note that this guarantees the greatest flexibility in composing models of the stent graft—whether they may be commercial or custom ones. The following chart summarizes the approach:

1. Define the stent's properties (only once per stent graft).
2. Initialize the first stent segment's approximate pose.
3. Register the segment to the image to find correct pose and deformation.
4. Guess approximate pose of the next segment.
5. Repeat steps 3 and 4 until the last segment is reached.

Note that only the first two steps require user input while the other ones run automatically. Thus, we can call it a *semi-automatic* approach. Admittedly, the objective of full automatization—which should always be a long-term goal of medical image analysis—is not reached, yet. However, some ideas on that issue are presented in section 10.2.

[1] The definition of the stent's properties is done by a simple XML file. See section A.2 in the appendix for the XML model specification.

6.4.2. Motivation

The initial situation of the approach described above is that we want to have a model that is used for the stent graft detection. We motivate this choice by the fact that we can incorporate a priori knowledge on the object shape into the model which yields improved segmentation results concerning accuracy and robustness (in contrast to methods that rely solely on image intensities). Furthermore, there are other situations, where the model might come in handy: In the future, a cooperation between computer scientists and engineers will be established in order to examine the influence of blood flow on the stent graft by fluid simulations for which a model of the device is needed, too. The one developed here certainly needs some extensions for this purpose (for more realistic physical behavior) but it could provide a good starting point.

Once the choice on using a model-based approach is made, and concerning the solution options summarized in section 6.3 we can finally conclude that the method of *segmentation via registration* is the most suitable one for the task at hand because it allows us best to make use of a priori knowledge via our geometrically and physically inspired model. *Nonrigid* registration is necessary because we are not only required to find the model's pose but also shape deformations. It should be noted, however, that these changes in shape are not undergoing arbitrary deformations but are always restricted to the nature of the model, i.e. the modeled parameters that affect its shape.

6.4.3. Summary

Let us summarize the tasks and requirements to accomplish the goal of detecting the stent with the approach described in the last section.

Tasks. There are three basic tasks to be performed in order to realize the outlined approach:

1. Develop a model that captures the overall shape of the individual stent segments (including deformations) and establish a relationship between the segments to represent the complete stent.

2. Formulate the nonrigid registration problem and solve it. This usually requires an iterative numerical solution accomplished by *optimization*. Hence, an appropriate *objective function* has to be developed.

3. Possibly some preprocessing on the image data is required to facilitate or improve the detection.

Prerequisites. To wrap it up, there are also some prerequisites that are assumed to be given. These are mainly concerning the specific stent graft in use and as already stated are not a big deal to determine in advance:

- Dimensions of the stent segments (radii and heights),

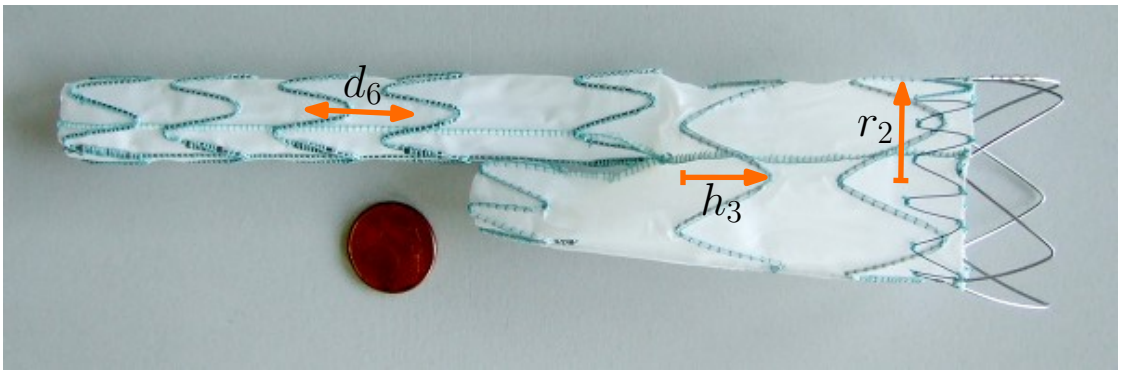


Figure 6.2.: Prerequisites for the registration process are the approximate dimensions (radii r_i and heights h_i) of the stent segments as well as the inter-segment distances d_i.

- Distances between the stent segments,
- Initialization of the first segment's pose.

These values have to be specified only once for each stent graft prior to the intervention or post-treatment and can be re-used for every monitoring routine (except for the approximate initialization of the first segment). Please, take a look at figure 6.2 which illustrates the parameters that are to be measured from the stent graft. Note that only *approximate* values are required, because the registration process determines the correct ones according to the given image.

Now that we have an overview of our approach and the tasks associated with it we can proceed to some general formalizations we need to develop the required models and methods.

Part III.

Methodology

7. Fundamentals and Formalizations

This chapter takes care of some fundamental methods and formalizations that are required to develop and describe the approach used in this thesis. Formal definitions related to digital images and image segmentation are followed by some general introductions to registration and optimization including the associated mathematical foundations.

7.1. Digital Image and Image Segmentation

The general aspects of digital imaging and some common segmentation methods have already been addressed. Hence, let us focus on some formalizations which will need at a later point.

7.1.1. Formal Representation of a Digital Image

Section 2.1.1 already introduced the representation of a digital image as a discrete grid that contains discrete values. For grayscale images, this actually describes a mapping of the 2D or 3D image space to one single intensity value. Formally, such a mapping is defined by

$$I : D \rightarrow \mathbb{R};\ D \in \mathbb{N}^d \tag{7.1}$$

with d being the image *dimension* (2 or 3 in our case), and *data points* $\mathbf{x} \in \mathbb{N}^d$ (pixels for $d = 2$, voxels for $d = 3$). D is called the image *domain*. Note that this is still a continuous definition although discrete values are used in practice for the intensity values (e.g. $0, \ldots, 255$).

7.1.2. Formal Definition of Image Segmentation

A *segmentation* of an image $I : D \rightarrow \mathbb{R}$ into *regions* is a partition of its domain D into subsets $R_i, i = 1, \ldots, k$, such that

$$D = \bigcup_{i=1,\ldots,k} R_i. \tag{7.2}$$

One of these regions may be considered as *background*. Note that it is not necessarily required for these subsets to be disjoint, i.e. $R_i \cap R_j = \emptyset, i \neq j; i, j \in \{1, \ldots, k\}$, because one point may belong to different segmentations. For example, a point could be part of both a vessel segmentation and a liver segmentation, if the vessel in question belongs to the liver.

7.2. Introduction to Registration

When talking about *registration* one first has to clarify what is actually meant by this term. Frequently, it refers to *image-to-image registration* which dominates this field of research. However, there is another type, namely, *model-to-image registration*. There are further possibilities to classify registration and the most important ones are given in the following section. Generally speaking, the purpose of registration is the alignment of data. In the sense of image-guided interventions this is related to image data as well as anatomical models or models of medical devices which are supposed to get aligned to each other or to the real world.

Let us first try to classify the various types of registration before addressing the solution options for registration problems and relating our approach to these issues.

7.2.1. Classification of Registration

There may be several ways to classify the broad field of registration. Some important possibilities regard the *subject* of the registration, the involved *transformations*, the given image or model *dimensions*, and the *type of data* that is utilized.

Image-to-Image vs. Model-to-Image Registration. As already mentioned, there are basically two different types of registration. *Image-to-image registration*, or simply *image registration*, denotes the process of aligning two (or several) images in a common coordinate frame in a way that corresponding structures can be overlaid and visualized. The involved images do not necessarily be of the same type but frequently include different imaging modalities. A typical example of image registration would be to register CT volumes (commonly used for visualizing bone structure) with MRI volumes (typically used for soft tissue) in order to combine the specific strengths of both modalities in visualizing internal body structures. Another example is the registration of preoperative CT scans with interventional 2D images in order to enhance the latter by a third dimension. Beyond the registration *across modalities* for the *same patient*, the procedure is used for the examination of *time series* for the same patient, as well as *across subjects* for the creation of image atlases.

Model-to-image registration, on the contrary, is a way of fitting a predefined model to an image such that the model corresponds (as good as possible) to the object of interest visible in the image. The model can be determined in various ways: It could be given as a pixel/voxel cloud from a prior segmentation or may be a statistical model learned from several segmentations of a set of images (see the paragraph on statistical shape models in section 6.3.3). In our case, however, the model is defined parametrically by a set of mathematical functions. Section 8.1 explains how this is done.

Rigid vs. Nonrigid Registration. Registration can also be classified in terms of the transformation involved. *Rigid* registration means that images or models are registered using a *rigid transformation*, i.e. only *translation* and *rotation* are estimated

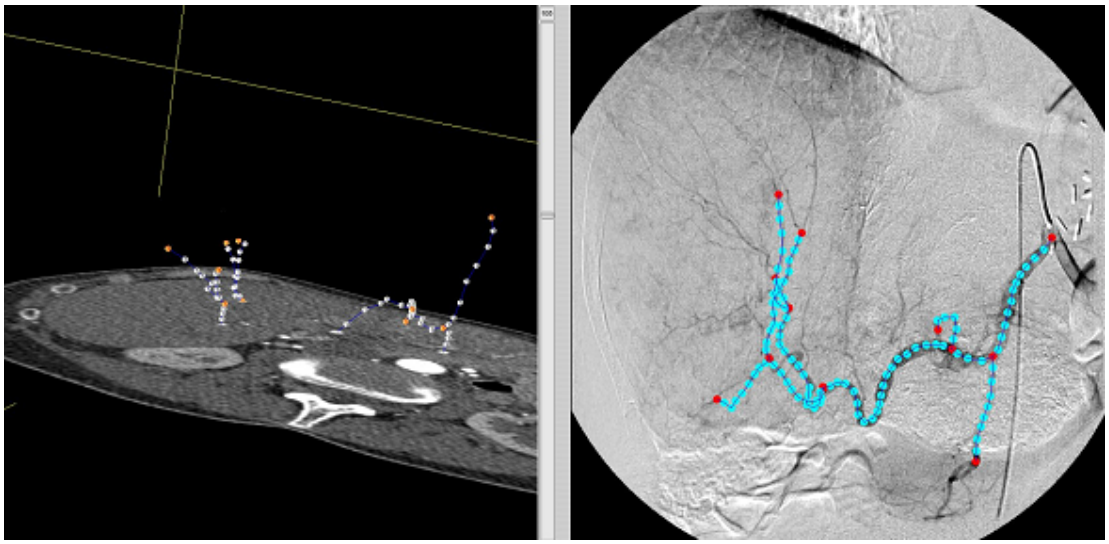


Figure 7.1.: Feature extraction provides points from an interventional 2D image (right) and a vessel segmentation from a preoperative volume (left), which are used for feature-based vessel registration. (Images courtesy of Martin Groher, CAMPAR, TUM.)

giving the *pose* (*position* and *orientation*) of the model in the image or relating one image to another, respectively. Deformations of the images or the model shape are not taken into account here. Consequently, this type of registration can only be used if such deformations are not expected to occur.

Nonrigid (or *deformable*) registration does account for these variations by estimating further parameters that affect the model shape or image distortion. Consequently, the problem is harder to solve as the dimension of the parameters to estimate increases.

2D-3D vs. 3D-3D Registration. Talking about dimensions, we can classify registration by the model and image dimensions involved. In the medical practice, either *3D-3D* or *2D-3D registration* is typically utilized. The former is needed for the alignment of complementary images (see the CT/MRI example above), for the registration of interventional 3D images to the real world or for fitting 3D models to volumes.

2D-3D registration is mainly required for the alignment of interventional 2D images to the real world (3D), for the registration of (usually preoperative) volumes to interventional 2D images or for fitting 3D models to 2D image data.

Feature-Based vs. Intensity-Based Registration. We can also make a distinction concerning the actual data the registration utilizes. *Intensity-based* registration only uses *intensity values* in the images to align opposed to *feature-based* registration which tries to relate *image features*, i.e. *geometrical data*. The latter typically are point

clouds, contours or surfaces that may be extracted by a prior segmentation (see figure 7.1 for an example).

As a matter of fact, feature extraction, in turn, has to be based on intensity values somehow because this is obviously the only information we get from an image. The actual registration purely acts on the extracted features, though. It is important to note that the two approaches can be mixed which has in fact been done for various applications.

7.2.2. Solution of a Registration Problem

Only rare circumstances allow for a direct *analytic* solution of a registration problem. This may be the case where exact point or feature correspondences are present that can be directly related. For example, markers or anatomical landmarks visible in the images could provide such point correspondences.

Many registration problems, however, lack these direct correspondences and, hence, a closed form solution cannot be derived. These cases require a *numerical* solution which is usually accomplished in an *iterative* fashion via *optimization*. There, the problem is addressed by *minimization* or *maximization* of a *cost function* (the *objective function*) which is directly related to image intensity values or image features and the parameters to estimate.

7.2.3. Summary

As our approach concerns, we are in the domain of *nonrigid model-to-image registration* since we want to relate a model of the stent to image data in order to *localize* it and determine its *shape* which is up to deformation. There is to mention, though, that the latter is not an arbitrary deformation but stays in the range defined by the model parameters, that is to say, the model points are always bound to a deformable parametric curve (see section 8.1).

Dealing with both 2D and 3D image data and a 3D model we address *3D-3D registration* as well as *2D-3D registration*. As we will see, the latter is basically done by projecting the 3D model to the 2D image space. Nevertheless, optimization acts on the parameters of the 3D model and, hence, we can really speak of 2D-3D registration.

The stent model is defined parametrically, i.e. as a continuous curve. As explained later (in section 8.1.3) the curve is *digitized* such that we get a point set representing the model which can be related to discrete image points (pixels or voxels). Although we have features here—namely, the model points—the objective function used for optimization is related to image intensity values. Consequently, we may classify our approach as *intensity-based registration*. Nonetheless, the method may be modified to utilize extracted contours—but this is up to future work.

7.3. Introduction to Optimization

Generally, *optimization* denotes the process of finding best element from a set of available alternative subject to some criteria. In mathematical terms, an *optimization problem* consists of a set of independent variables or *parameters* and a measure of "goodness" defined by an *objective function* (also called *cost function*, *energy function*, or *energy functional*). In addition, there may be certain conditions or restrictions, the *constraints*, that define acceptable values of the variables. The *solution* of an optimization problem is a set of allowed values of the parameters for which objective function assumes an "optimal" value. Mathematically, this involves *maximizing* or *minimizing* the objective function.

In the simplest case, we may wish to minimize or maximize a real function. In more challenging situations, however, there could be several objectives we want to account for and, thus, we also have several (usually conflicting) objective functions. In this case, *multi-objective optimization* is necessary.

There are many real-life situations that can be modeled as optimization problem. This is convenient if we want to maximize some profit or minimize some weight, for example. A typical situation of multi-objective optimization in engineering arises when one wants to maximize performance while minimizing fuel consumption of a vehicle. As our registration problem concerns we want to find the pose and shape of our model from the set of possible configurations such that the model matches best the actual stent in the given image.

7.3.1. Formal Definition

In our case, it is sufficient to have a look at problems that can be posed in terms of a single real-valued scalar objective function. Formally, such an optimization problem can be defined as

$$\begin{aligned} \min_{\mathbf{p}\in\mathbb{R}^n} f(\mathbf{p}) \quad \text{subject to } & c_i(\mathbf{p}) = 0,\ i = 1, 2, \ldots, m'; \\ & c_i(\mathbf{p}) \geq 0,\ i = m' + 1, \ldots, m, \end{aligned} \tag{7.3}$$

with the objective function $f : \mathbb{R}^n \to \mathbb{R}$, the set of parameters stacked up in vector $\mathbf{p} \in \mathbb{R}^n$, and real-valued scalar constraint functions $\{c_i\}$, the *problem functions*. The domain $\mathbb{R}^n$ of f is called the *search space* or the *choice set*, while its elements are denoted as *candidate solutions* or *feasible solutions*. A feasible solution that *minimizes* the objective function is called an *optimal solution*.

Although it may be the goal to compute the minimal value of the objective function, $f(\mathbf{p}) = \min_{\mathbf{p}\in\mathbb{R}^n} f(\mathbf{p})$, we are usually interested in the actual optimal solution. In other words, we are seeking an element $\hat{\mathbf{p}} \in \mathbb{R}^n$ such that $f(\hat{\mathbf{p}}) \leq f(\mathbf{p}), \forall\, \mathbf{p} \in \mathbb{R}^n$. This is commonly formalized as:

$$\hat{\mathbf{p}} = \operatorname*{argmin}_{\mathbf{p}\in\mathbb{R}^n} f(\mathbf{p}). \tag{7.4}$$

Note that we just described *minimization*—the contrary process of *maximization*, however, is done in an analogous manner.

7.3.2. Solution of an Optimization Problem

Now, the objective is to compute the optimal solution of the optimization problem. There may be the case, that besides a *global optimum*, there are several *local optima*. Starting from an initialization, our objective may either be to find the global optimum (*global optimization*) or one of the local ones (*local optimization*). The goal of reaching the global optimum is a very hard mathematical problem, because it can happen that on our way to the global optimum we get stuck in a local one. The complexity of this problem is highly dependent on the topology of the objective function. Thus, we may redefine our objective such that we at least want to find the optimum in question given a reasonable initialization.

Mathematically, a point $\mathbf{p}^*$ is a local minimum if

$$\exists\, \delta > 0 : \forall\, \mathbf{p} \in \mathbb{R}^n, \|\mathbf{p} - \mathbf{p}^*\| \leq \delta : f(\mathbf{p}^*) \leq f(\mathbf{p}); \tag{7.5}$$

that is to say, on some region around $\mathbf{p}^*$ all of the function values are greater than or equal to the value at that point. By the way, local maxima are defined analogously.

The common practice to solve an optimization problem is to make use of existing methods in order to solve the problem dependent on the nature of the problem, i.e. the properties of $\{c_i\}$ and f (such as number of constraints, linear/nonlinear objective or constraint functions) along with the desired accuracy and computational costs.

7.3.3. Classification of Optimization Methods

Optimization methods can basically be classified in two categories: *Deterministic* methods try to explore the search space in an efficient deterministic way, while *probabilistic* methods (such as *Monte Carlo*-based approaches) use some heuristics in order to choose feasible solutions to test. It is further possible to divide the deterministic approaches into different categories: *Gradients-based* methods (like the *Newton* and *gradient-descent* method) make use of first and possibly second order derivatives of the objective function in order to approach the optimum. *Gradient-free* methods (mainly *direct search* methods like *downhill simplex*), on the contrary, do not require computation of derivatives. There may be further ways to classify the different methods, but details are spared here. For more information on optimization see the relevant textbooks (e.g. [GMW82] and [QSS06]).

8. Methods and Models

In this chapter the used methods are described and formalized along with a detailed description of the model that has been developed in order to detect the stent graft in 2D and 3D image data. First, a geometrically and physically inspired *parametric model* of the stent—or rather its components, the individual segments—is presented as the object of the registration process. However, the model not restricted to this application but may also be used for other tasks at hand. Afterwards, the details of the model-to-image registration approach are explained and a further section focuses on *optimization*—with emphasis on the *downhill simplex method*—as common numerical means to solve the registration problem. There, the required *objective function* that has been designed for this purpose is presented—or rather two different versions of it. In the end, a few words are spent on the utilized *preprocessing* methods that may be either a prerequisite to the stent detection or yield an improvement of the detection robustness. That section addresses topics like *distance transform* and *background removal*.

8.1. Stent Segment Model

8.1.1. Modeling a Stent Segment

By looking at the pictures (4.2(a) and 4.2(b)) and CT scans (9.1) of the stent graft one can easily identify the individual segments it is composed of. We are actually modeling only the stent-part of the stent graft, i.e. the metal structure that is clearly visible in radiographic images. The tubular sine-shaped segments are predestined for parametric modeling. Hence, we define the stent segment model to be the curve,

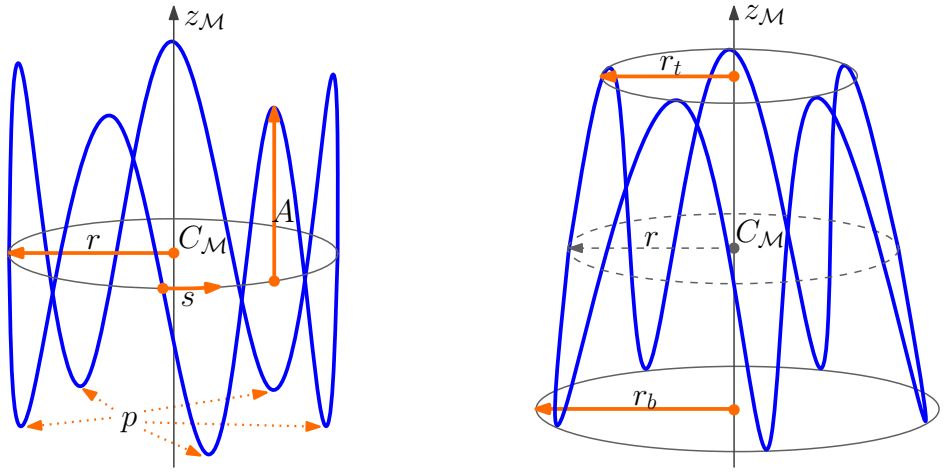
$$m_0(t) = (x_0(t), y_0(t), z_0(t)) \,, \tag{8.1}$$

consisting of the set of parametric equations

$$x_0(t) = r\cos(t) \tag{8.2}$$
$$y_0(t) = r\sin(t) \tag{8.3}$$
$$z_0(t) = A\sin(pt + s) \tag{8.4}$$

with the free parameter t and five constants: Radius r, amplitude A, period p and phase shift s.

The radius defines the segment's width, the amplitude specifies the segment's height, the period is equal to the number of peaks (on either side) and the phase shift merely



(a) Stent segment model.

(b) Extended stent segment model.

Figure 8.1.: (a) Stent segment model with $p = 5$ and the parameters A (amplitude), r (radius) and s (phase shift). (b) The extended model with r_t (top radius) and r_b (bottom radius). Note that the central radius r is given by $r = \frac{r_t + r_b}{2}$.

shifts the starting point, which has the effect of rotating the model around its central axis $z_{\mathcal{M}}$. We will later see that the phase shift can be ignored[1] in our case as rotations of the model are handled in a more general way by the pose (given by equation 8.29 in section 8.2.2). Figure 8.1(a) visualizes the model and its associated parameters.

Note that this model specifically applies to stent grafts used for EVAR. Nevertheless, with little adjustments other types of stents and stent grafts may be represented by a similar model.

8.1.2. Extensions to the Model

The model as defined above is not capable of dealing with arbitrary deformations of stent segments. Deformations in a general sense are not taken into account so far and may be up to future work. However, one adjustment has been made in order to cope with situations where one end of a segment has extended to a higher degree than the other one. This usually occurs while releasing the stent graft within the aorta or in case it gets deformed by the blood vessel where it narrows. Thus, the stent segment model needs to have a varying radius that is dependent on the z-position on the model. This can be achieved by introducing two radii—a top radius r_t and a bottom radius r_b—and a function $r_s(t)$ to establish the required relationship. Thus, the extended stent segment model,

$$m_s(t) = (x_s(t), y_s(t), z_s(t)) \,, \tag{8.5}$$

[1] We just leave the phase shift s here for the sake of completeness of the model. It can assumed to be zero for our purpose, though.

is composed of the set of parametric equations

$$x_s(t) = r_s(t)\cos(t) \tag{8.6}$$
$$y_s(t) = r_s(t)\sin(t) \tag{8.7}$$
$$z_s(t) = A\sin(pt+s) \tag{8.8}$$

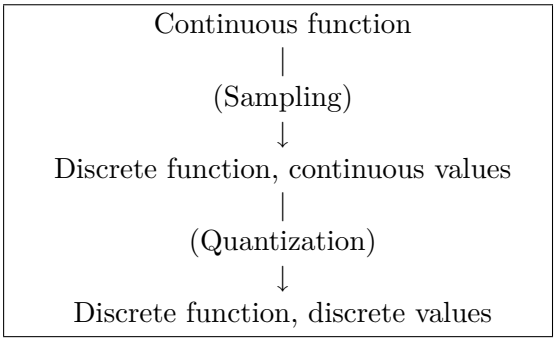
where

$$r_s(t) = \left(1 - \frac{\sin(pt)+1}{2}\right) r_t + \frac{\sin(pt)+1}{2}\, r_b. \tag{8.9}$$

Figure 8.1(b) shows the final extended model $m_s(t)$ which is used in the following.

8.1.3. Model Digitalization

The equations given in the previous chapter define the functions that make up the stent segment model. In order to make it suitable for implementation on digital computers and to be able to compare the continuous curve with discrete 2D or 3D image points we need to *digitalize* the model. This process is common in *digital signal processing* but may be applied to continuous models (instead of continuous time signals) as well. *Digital geometry* deals with discrete point sets considered to be digitized models or images of objects of the 2D or 3D Euclidean space. The digitalization of continuous objects is known as *digitizing* and involves *sampling*, or *discretization*, and *quantization*. We refer to discretization as the process of taking samples of the model at given intervals whereas quantization turns the function values at the sample points into discrete ones. The effect of these two processes is visualized in figure 8.3 and illustrated by the following chart:



Formalization of the Sampling Process. Formally, the sampling process can be modeled by using the *Dirac comb* (or *Sha function*) which is defined as

$$\Delta_T(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT) \tag{8.10}$$

with the *sampling period*, or *sampling interval*, T.

It is composed of a sequence of *Dirac impulses* defined by the *Dirac delta function*[2]

$$\delta(t) = \begin{cases} 1, & t = 0 \\ 0, & t \neq 0. \end{cases} \tag{8.11}$$

Applying the Dirac comb to the model $m_s(t)$ yields

$$m_d(t) = m_s(t)\Delta_T(t) = m_s(t) \sum_{k=-\infty}^{\infty} \delta(t - kT) = \sum_{k=-\infty}^{\infty} m_s(kT)\delta(t - kT) \tag{8.12}$$

with sampling period T as given by equation 8.18.

The function $m_d(t)$ is still a continuous curve. However, its values are unequal to zero only at discrete points kT. Thus, we are able to write it as *discrete* sequence

$$m_d[k] = \sum_{k=-\infty}^{\infty} m_s(kT)\delta[k]. \tag{8.13}$$

Using the fact that the model is a periodic curve we actually need to sample it only in the interval $\mathcal{I} = [0, 2\pi[$ which yields the discrete *finite* sequence

$$m_d[k] = \sum_{k=0}^{n} m_s(kT)\delta[k];\ n = \left\lfloor \frac{2\pi}{T} \right\rfloor. \tag{8.14}$$

Estimation of the Sampling Period. Now that we know how to sample our model there is still one open question: How shall we choose the sampling period in order to get a nice and dense discrete representation of the model in the image? We are seeking to neither *under-sample* nor *over-sample* the curve. Under-sampling, on the one hand, would lead to inaccuracies while over-sampling, on the other hand, would result in high costs for further computations.

As the model size varies the sampling rate has to be dependent on the model's radius and amplitude. We can estimate the number of required samples by the lengths of the lines joining the segment peaks and middle points—i.e. the curve's inflection points P_{0_i} and local extrema P_{p_i}. Figure 8.2 actually makes that much clearer than words can do.

We are interested in the length of the line segment $l = \overline{P_0 P_p}$ as approximation of the length of the curve between these two points. Note that $\overline{P_0 P_c}$ and $\overline{P_c P_p}$ meet at a right angle which means that l can be derived using the *Pythagorean theorem*:

$$l^2 = A^2 + c^2 \Leftrightarrow l = \sqrt{A^2 + c^2}. \tag{8.15}$$

The line segment $c = \overline{P_0 P_c}$ is actually a chord lying on the circle defined by radius[3] r

[2] Note that this is merely a heuristic definition and not a mathematical one because the Dirac delta is not a true function.

[3] If the segment model is made up by two radii (top and bottom radius) the central radius is given by

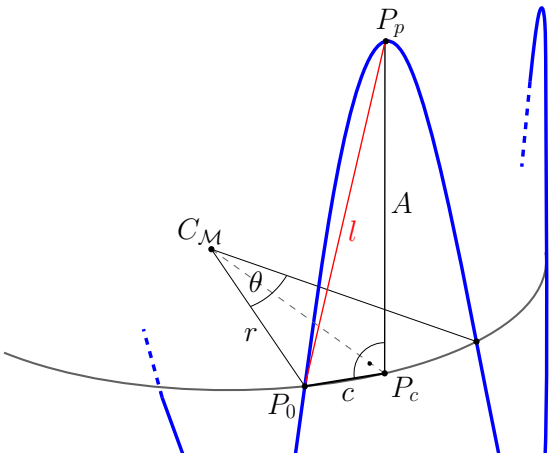


Figure 8.2.: Section of the stent segment model. We are interested in l (red line) as estimation for the length of the curve between P_0 and P_p. It is needed to compute the sampling frequency used to sample the model.

and, thus, its length is given by

$$c = r \operatorname{crd}\left(\frac{\theta}{2}\right) = 2r \sin\left(\frac{\theta}{4}\right);\ \theta = \frac{2\pi}{p}. \tag{8.16}$$

For one peak the length l is needed *twice* and the number of peaks on *either* side is given by the period p. This makes a total of $4p$ lines spanning the complete stent segment. Using this we can finally derive the sampling frequency

$$f_s = 4pl = 4p\ \sqrt{A^2 + c^2} = 4p\ \sqrt{A^2 + 4r^2 \sin^2\left(\frac{\pi}{2p}\right)}. \tag{8.17}$$

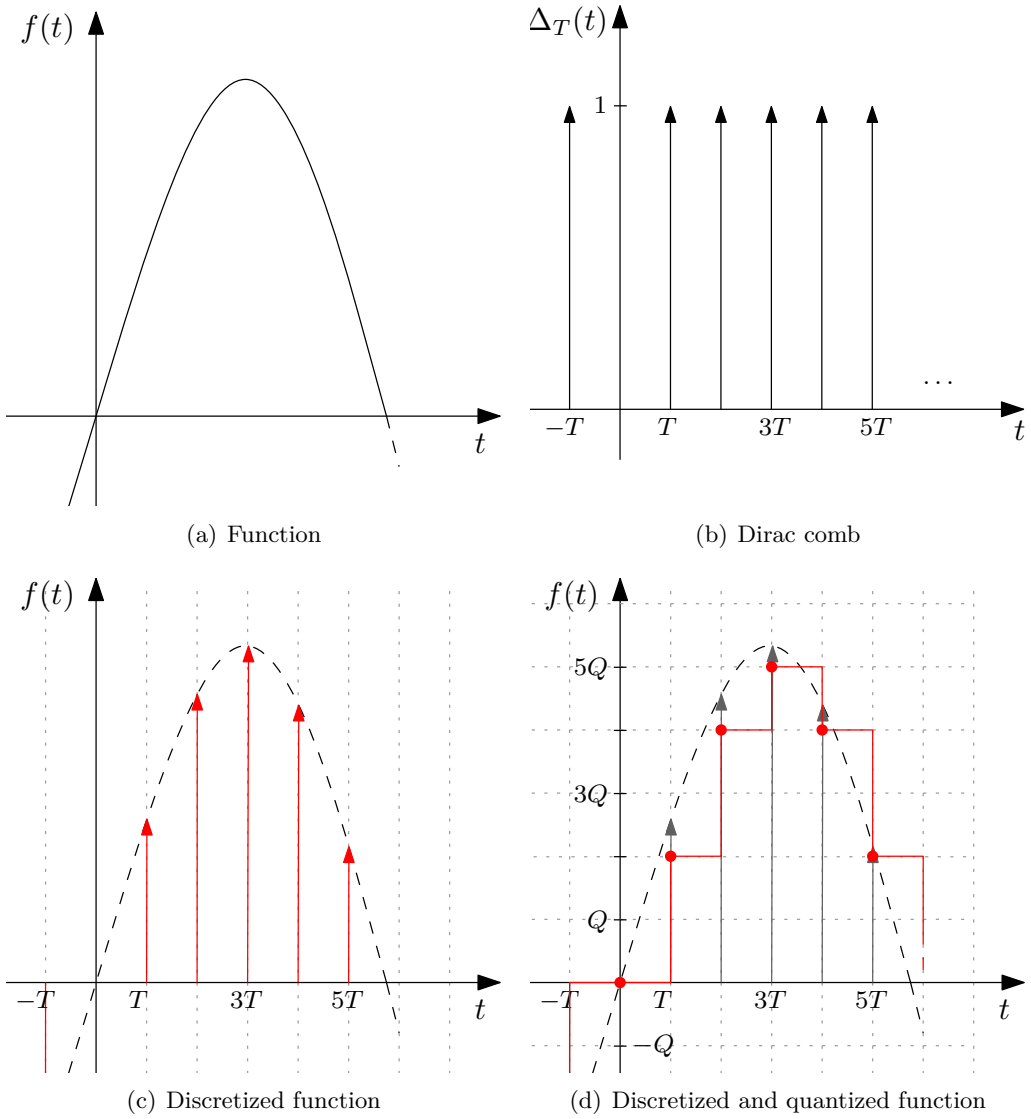
The *sampling period* (or *sampling interval*) T is simply the inverse of f_s:

$$T = \frac{1}{f_s}. \tag{8.18}$$

Sampling the model $m_s(t)$ in intervals T gives us a decent number of samples. The idea behind this choice is that for a curve between P_0 and P_p whose length is estimated by l we need about l pixels (or voxels, resp.) to sample it properly.

Quantization of the Sampled Points. Now there is just one step left, namely, the transition from points given in real numbers to image voxels which are commonly defined as integer coordinates (x, y, z) on a voxel grid. This can be achieved by *quantization*—for example, by rounding the values of $m_d[k]$ to the nearest integer. However, referring

the mean value $r = (r_t + r_b)/2$.



(a) Function

(b) Dirac comb

(c) Discretized function

(d) Discretized and quantized function

Figure 8.3.: Sampling and quantization of a curve. The continuous function $f(x)$ (a) is multiplied by the Dirac comb $\Delta_T(t)$ (b) which yields the discretized function $f_d(t)$ (c) only defined at points iT. Quantization rounds or truncates these samples to values jQ and gives $f_q(t)$ (d).

to [PH04] (section 7.1.7, "Understanding Pixels") we simply truncate the continuous coordinates to discrete ones.

A general scalar quantization function is of the form

$$Q(x) = g(\lfloor f(x) \rfloor) \tag{8.19}$$

with arbitrary real-valued functions $f(x)$ and $g(x)$. The integer result $i = \lfloor f(x) \rfloor$ is referred to as the *quantization index*. Using this definition enables modeling a quantization with different *quantization step sizes*. Formally, the process actually describes an approximation of a continuous ranges of values by discrete values by mapping intervals I_i to numbers c_i:

$$I_i = [a_i, b_i[\rightarrow c_i;\ I_i \subset \mathbb{R}, c_i \in \mathbb{N}. \tag{8.20}$$

Since $\mathbb{R}$ can be partitioned into disjoint intervals I_i as

$$\mathbb{R} = \bigcup_{n \in \mathbb{N}} I_n;\ \forall\, I_i, I_j \subset \mathbb{R} : I_i \cap I_j = \emptyset; i, j \in \mathbb{N}, \tag{8.21}$$

the quantization function is a mapping

$$Q : \mathbb{R} \rightarrow \{\ldots, c_{-2}, c_{-1}, c_0, c_1, c_2, \ldots\};\ c_i \in \mathbb{N}. \tag{8.22}$$

In our case we simply want to truncate the values to integers in a *uniform* way (meaning we use a uniform step size). Hence, we simply use the quantization function

$$Q_u(x) = \lfloor f(x) \rfloor \tag{8.23}$$

which maps intervals I_i to integers i,

$$I_i = [i, i+1[\rightarrow i;\ I_i \subset \mathbb{R}, i \in \mathbb{N}, \tag{8.24}$$

resulting in[4]

$$Q_u : \mathbb{R} \rightarrow \{\ldots, -2, -1, 0, 1, 2, \ldots\}. \tag{8.25}$$

The quantization function given above is a *scalar* function (opposed to a vector function) but we have a parametric curve. However, this is not a problem since we can easily apply the quantization to the individual parametric equations by defining

$$Q_s(m_s(t)) = \lfloor m_s(t) \rfloor = (\lfloor x_s(t) \rfloor, \lfloor y_s(t) \rfloor, \lfloor z_s(t) \rfloor) \tag{8.26}$$

which quantifies our curve uniformly in every dimension.

[4] The quantization is still in an infinite range which cannot be represented in a digital system. However, without loss of generality we can silently assume that we are only acting in a range given by the image dimensions, i.e. in intervals $I_x = [0, d_x], I_y = [0, d_y], I_z = [0, d_z]$ with d_x, d_y, d_z being the image dimensions in x-, y-, and z-direction, respectively.

Together with equation 8.5 we can give a discrete and quantized stent segment model

$$m_q[k] = Q_s(m_d[k]) = \lfloor m_d[k] \rfloor = \sum_{k=0}^{n} \lfloor m_s(kT) \rfloor \delta[k] = \\ = \sum_{k=0}^{n} (\lfloor x_s(kT) \rfloor, \lfloor y_s(kT) \rfloor, \lfloor z_s(kT) \rfloor) \delta[k]. \tag{8.27}$$

For convenience, we write this sequence as *set*[5] of 3D vectors and define the final stent segment model in voxel space as

$$\mathcal{M} = m_q[k] = \{\mathbf{m}_0, \mathbf{m}_1, \ldots, \mathbf{m}_n\} \tag{8.28}$$

with discrete points $\mathbf{m}_i = (x, y, z)^{\mathrm{T}} = (\lfloor x_s(iT) \rfloor, \lfloor y_s(iT) \rfloor, \lfloor z_s(iT) \rfloor)^{\mathrm{T}} \in \mathbb{N}^3$. Of course, from a computational point of view the points $\{\mathbf{m}_i\}$ are defined only in a finite subset of $\mathbb{N}^3$.

Figure 8.3 summarizes the whole sampling and quantization (8.3(d)) process for an exemplary continuous function $f(x)$ in 2D (as this is easier to visualize than for our 3D model)[6].

Note that, although not always explicitly declared, we may assume all subsequent operations and transformations to the model points to be carried out in discrete space (resulting in discrete points again).

8.2. Model-to-Image Registration

The equations given in the previous section define the points that make up the stent segment model. In order to place the model in the 2D or 3D image space a transformation is necessary to map the points from the model coordinate system $C_{\mathcal{M}}$ to the image coordinate system C_I. Using this transformation and the model parameters we are able to perform the registration in order to estimate both pose and shape of a stent segment. As outlined in the overview of our approach (section 6.4.1) we perform the registration in a piecewise manner starting from one initial segment and registering the subsequent ones iteratively in order to find all the segments the stent graft is composed of. Let us summarize the procedure again before modeling the transformations and giving a formalization of the registration problem.

8.2.1. Piecewise Registration of the Segments

The piecewise registration process of the individual stent segments is illustrated by figure 8.4 and the following pseudo-code:

[5] We may use a set instead of a *tuple* because our definition of the sequence $m_q[k]$ (equation 8.27) guarantees that every point is unique.

[6] You might realize that, indeed, the function $f(x)$ in figure 8.3 resembles a peak of the stent segment model so you may think of it being one—just flattened to 2D.

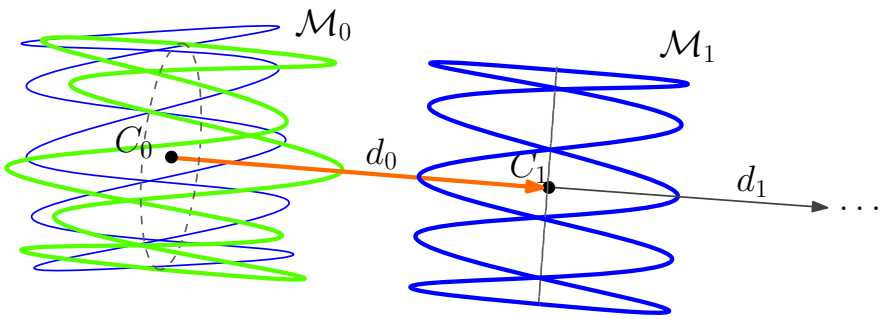


Figure 8.4.: Piecewise registration of the stent segment model (in 2D): Starting with a rough initialization of the first segment $\mathcal{M}_0$ (left blue curve) the correct pose and shape it determined via registration (green curve). Going along the axis from its center C_0 by the predefined distance d_0 (according to model specification) the center C_1 of the next segment $\mathcal{M}_1$ is estimated, and a further registration process starts using the standard shape (right blue curve). The iteration proceeds until the last segment has been registered to the image.

Algorithm 8.1. *Piecewise segment registration.*

```
1 Initialize the pose of the first segment (via user input);
2 Initialize the first segment with the standard shape (according
    to model specification);
3 Register the first segment to get the correct pose and shape;
4 FOR (segment 2 to n):
5   Compute the previous segment's axis (from position and rotation);
6   Estimate the center of the current segment (using the given
      inter-segment distance);
7   Initialize segment with estimated position as well as standard
      rotation and shape (according to model specification);
8   Register the segment to get the correct pose and shape;
9 END FOR.
```

Interactive user input is only required in the first step in order to initialize[7] the approximate pose of the first segment. For the 2D case, only position and rotation about the image plane normal are required while the other rotations are initialized with the standard configuration (i.e. values set to zero). Note that this also applies to the initialization in step 7. There, the rotation about the image plane normal is initialized according to the previous segment. In 3D, all rotations are taken from the

[7] For the 2D case, initialization is done by clicking four points that outline the stent segment, and for 3D, a couple of controls for position and rotation are implemented in the application's GUI (see section A.1 for screenshots).

previous segment. The initial shape (i.e. amplitude and mean radius) of a segment and the inter-segment distances are predefined in the model specification[8] which is assumed to be given for the stent graft at hand.

8.2.2. Model Pose in the Image

Whereas the model shape has already been formalized in section 8.1, its pose has not been addressed, yet. Let us first define it in 3D space before approaching the 2D case.

3D Pose of the Model. The segment model's *pose*—i.e. the *position* and *orientation* of the segment in 3D space (or a volume)—can be defined by a *rigid transformation* of the model points $\{\mathbf{m}_i\}$ (as specified by equation 8.28), which transforms the original points (defined in the model coordinate system $C_{\mathcal{M}}$) to new points (in the image coordinate system C_I).

In general, such a transformation is given by

$$\mathbf{x}' = \mathrm{R}\mathbf{x} + \mathbf{t} \tag{8.29}$$

with a 3×3 orthogonal rotation matrix R and a translation vector $\mathbf{t} = (t_x, t_y, t_z)^{\mathrm{T}}$. The rotation matrix can be thought of composition of three independent rotations about the image axes

$$\mathrm{R} = \mathrm{R}_z\mathrm{R}_y\mathrm{R}_x = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \tag{8.30}$$

specified by the *Euler angles* ψ, θ, and ϕ.

The transformation given by equation 8.29 can also be written using a 4×4 *homogeneous transformation matrix* H_{E} which yields

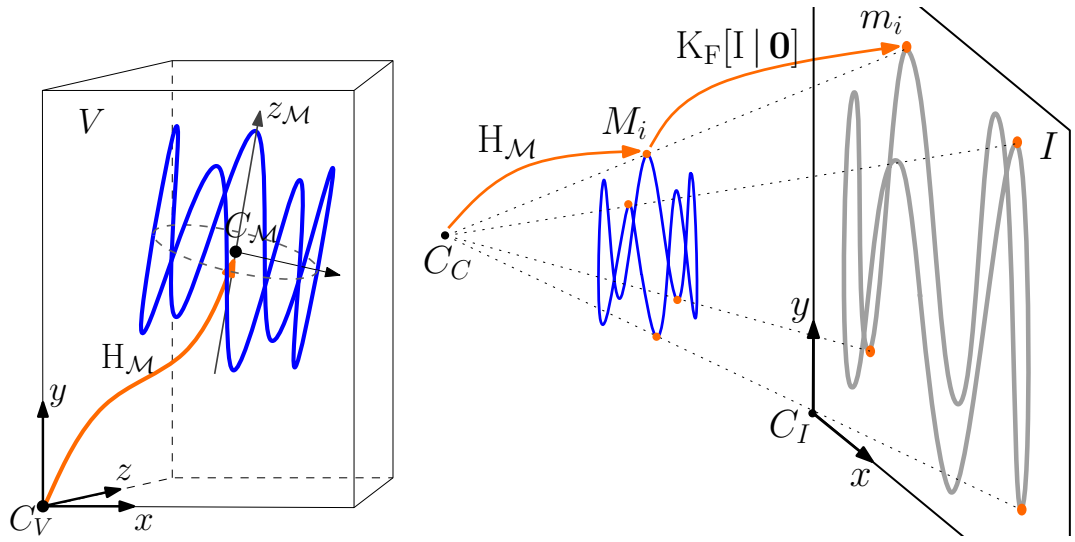
$$\mathbf{X}' = \mathrm{H}_{\mathrm{E}}\mathbf{X} = \begin{bmatrix} \mathrm{R} & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix} \mathbf{X} \tag{8.31}$$

with the homogeneous vectors $\mathbf{X} = (X, Y, Z, W)^{\mathrm{T}} = (\frac{X}{W}, \frac{Y}{W}, \frac{Z}{W}, 1)^{\mathrm{T}} = (x, y, z, 1)^{\mathrm{T}}, W \neq 0$, and $\mathbf{X}' = (X', Y', Z', W')^{\mathrm{T}}$; $\mathbf{X}, \mathbf{X}' \in \mathbb{P}^3$ (the *projective* space). H_{E} is actually a *Euclidean* transformation matrix, meaning it is solely composed of a rotation and translation in order to model a rigid motion.

In the following, the Euclidean transformation matrix that defines the stent segment model's pose in 3D space is denoted as $\mathrm{H}_{\mathcal{M}}$. Consequently, a model point $\mathbf{M}_i$ (in homogeneous coordinates) is mapped to

$$\mathbf{M}_i' = \mathrm{H}_{\mathcal{M}}\mathbf{M}_i. \tag{8.32}$$

[8] The model specification is given by an XML file (see section A.2).



(a) 3D pose of the model.

(b) 2D projection of the model.

Figure 8.5.: (a) The pose of the stent segment model in a volume V is given by the rigid transformation $\mathrm{H}_{\mathcal{M}}$ between the model coordinate center $C_{\mathcal{M}}$ and the volume origin C_V. (b) The model is projected onto a 2D image I via the camera projection $\mathrm{K_F}\,[\mathrm{I}\,|\,\mathbf{0}]$ with camera center C_C $(= C_W)$.

Figure 8.5(a) illustrates the transformation applied to the stent segment model.

Using this transformation the model can be placed at any location in a 3D image to overlay it with visible stent segments. Note that we do not model any scaling with the transformation matrix (which could also be done easily) since the model size is already defined by by the parameters r_t (top radius), r_b (bottom radius) and A (amplitude) (see section 8.1.2—esp. equation 8.5).

2D Projection of the Model. Modeling in 2D space requires a mapping of the given 3D model to 2D by using either an orthographic or perspective *camera projection*[9]. In general, this can be done by applying a 3×4 homogeneous *camera projection matrix* P:

$$\mathbf{x} = \mathrm{P}\mathbf{X}. \tag{8.33}$$

It is convenient to decompose[10] the projection P into an upper triangular *camera matrix* K, rotation matrix R and translation vector $\mathbf{t}$:

$$\mathrm{P} = \mathrm{K}\,[\mathrm{R}\,|\,\mathbf{t}]\,. \tag{8.34}$$

[9] A very good and detailed introduction to projective geometry in general and camera projections in particular is given in [HZ04]. See esp. chapter 6, "Camera Models".

[10] We can assume that a decomposition of P into its components is always possible. For details, please refer to [HZ04].

In this form it is easy to explain the effect of the projection: K contains the camera's *internal* parameters (such as focal length and skew) while R and $\mathbf{t}$ represent the camera's *external* parameters which relate the camera orientation and position to a world coordinate system. At this point we are only interested in the camera matrix K because it contains the projective information we need to map the model points to the image.

As an example of how the projection works—and how camera matrix and pose are related—let us consider the simple case of an *orthographic projection*:

$$\mathrm{P_O} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8.35}$$

$\mathrm{P_O}$ is a projection along the Z-axis which maps a (homogeneous) 3D point $(X, Y, Z, 1)^{\mathrm{T}}$ to the 2D image point $(X, Y, 1)^{\mathrm{T}}$ by simply dropping the Z-coordinate.

For a general orthographic projection this map may be preceded by a 3D Euclidean transformation $\mathrm{H_E}$ (as defined in the previous paragraph, see equation 8.31):

$$\mathrm{P_O}' = \mathrm{P_O H_E} = \begin{bmatrix} \mathbf{r}^{1\mathrm{T}} & t_1 \\ \mathbf{r}^{2\mathrm{T}} & t_2 \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix}, \tag{8.36}$$

with $\mathbf{r}^{i\mathrm{T}}$ referring to the i-th row of R.

Generally, radiographic imaging devices—which can also be seen as sort of camera—do not project an image orthographically. Instead, a *projective distortion* is typically involved which can be modeled by a *finite projective* camera with the camera calibration matrix

$$\mathrm{K_F} = \begin{bmatrix} \alpha_x & s & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix} \tag{8.37}$$

where α_x and α_x represent the focal length of the camera in terms of pixel dimension (in x and y direction, respectively) and $\tilde{\mathbf{x}}_0 = (x_0, y_0)$ is the *principal point*. The *skew* parameter s will be zero for most devices.

Wrapped up, we can give the projection of any medical imaging device producing projective images by

$$\mathrm{P_F} = \mathrm{K_F}\,[\mathrm{I}\,|\,\mathbf{0}]\,\mathrm{H_C} = \mathrm{K_F}\,[\mathrm{R_C}\,|\,\mathbf{t}_{\mathrm{C}}]\,, \tag{8.38}$$

with an Euclidean transformation $\mathrm{H_C}$ giving the camera's pose in the world coordinate system (defined by rotation $\mathrm{R_C}$ and translation $\mathbf{t}_{\mathrm{C}}$), a 3×3 identity matrix I, and $\mathbf{0} = (0, 0, 0)^{\mathrm{T}}$. The projection of a point $\mathbf{X}$ onto the 2D image I is then given by

$$\mathbf{x} = \mathrm{P_F}\mathbf{X}, \text{ or simply} \tag{8.39}$$

$$\mathbf{x} = \mathrm{K_F}\,[\mathrm{I}\,|\,\mathbf{0}]\,\mathbf{X}, \text{ if } C_C = C_W, \tag{8.40}$$

with camera center C_C and world coordinate center C_W. Since we are free to define to define our camera center arbitrarily we can choose it to be coincident with the world coordinate center, and, hence, can use the latter representation.

Due to the fact that the images are projective, we also want to apply the projection given by the finite projective camera matrix (8.37) to the stent segment model in order to project it onto the image of interest under correct perspective distortions (as implied by the imaging device[11]). This is necessary because real objects (such as the stent segments) undergo this projection, too, when imaged via the corresponding device. Applying the same projection to the model makes sure that we can overlay the model with the image objects exactly.

In order to be able to place the model in the image we precede the projection by the rigid transformation $\mathrm{H}_{\mathcal{M}}$ as defined above. Finally, a model point $\mathbf{M}_i$ (in homogeneous coordinates) is mapped to the 2D image by

$$\mathbf{m}_i = \mathrm{K}_{\mathrm{F}} \left[\mathrm{I} \,|\, \mathbf{0}\right] \mathrm{H}_{\mathcal{M}} \mathbf{M}_i. \tag{8.41}$$

Figure 8.5(b) shows the effect of the projection.

8.2.3. Formalization of the Registration Problem

In order to register the model to the image we need to determine both its pose and shape. As outlined in section 6.4 and detailed in section 8.2.1 we do this in a piecewise manner by starting with the first segment and consecutively registering the other ones.

3D-3D Registration. The 3D pose of the segment model $\mathcal{M}$ is given by a rigid transformation $\mathrm{H}_{\mathcal{M}}$ (see equation 8.32 in the previous section). Its shape is determined by the parameters modeled with the parametric equations 8.6 and the following. Summarized, we can stack up all the parameters to estimate into a vector

$$\mathbf{p} = (t_x, t_y, t_z, \psi, \theta, \phi, A, r_t, r_b)^{\mathrm{T}} \in \mathbb{R}^9, \tag{8.42}$$

where the first six parameters define the pose—translation and rotation (given by Euler angles)—and the last three the shape. Note that we neither estimate the phase shift (which already modeled by the rotations) nor the period (which we assume to be predefined). As a whole we have 9 parameters making a total of 9 degrees of freedom—six for the pose and three for the shape.

Imagine that we have given the segmentation of a stent segment in an image I, i.e. a region $R_S \subseteq I$ that contains all points that belong to the corresponding segment. Ideally, we want the model points to be identical with the actual segmentation, i.e. $\mathcal{M} = R_S$—or at least, $\mathcal{M} \subseteq R_S$, because the actual segmentation is likely to contain

[11] The projection that has been used by a medical imaging device to produce a specific image is usually stored in the system. Nevertheless, if the projection is impossible to determine it may always be reconstructed from four image points.

more points than the model (which is just a curve with line width one as opposed to the actual point cloud that may be thicker).

However, in general this will not be possible due to outliers of the model not matching with the segmentation because of deformations we did not account for. Hence, the objective is to estimate the model parameters such that as many points as possible are matching. Formally, this can be expressed by

$$\hat{\mathbf{p}} = \underset{\mathbf{p}\in\mathbb{R}^n}{\operatorname{argmin}} \sum_{\mathcal{M}} R_S(T(\mathcal{M}, \mathbf{p})), \tag{8.43}$$

with the function $T(\mathcal{M}, \mathbf{p})$ that transforms[12] the model $\mathcal{M}$ by application of the parameters $\mathbf{p}$, and the optimal solution $\hat{\mathbf{p}}$.

2D-3D Registration. Until now, we just have considered the 3D-3D registration problem. The 2D-3D registration is almost identical, though—with the only difference that we have a 2D image and, hence, a 2D segmentation $\overline{R}_S$. In section 8.2.2 we already examined how to project our model to the 2D space using the projective camera matrix defined by equation **??**. Hence, we just have to modify the function T and introduce a new function $T_{\mathrm{K}}(\mathcal{M}, \mathbf{p}, \mathrm{K_F})$ which does not only apply the parameters $\mathbf{p}$ to the model $\mathcal{M}$, but also the camera projection $\mathrm{K_F}$. Furthermore, we do not have to estimate a z-position, t_z, of the model in 2D space. Note that we still need three rotations, though. Thus, we also modify the parameters to estimate, $\mathbf{p}$, by just removing the unnecessary parameter which yields

$$\mathbf{q} = (t_x, t_y, \psi, \theta, \phi, A, r_t, r_b)^{\mathrm{T}} \in \mathbb{R}^8. \tag{8.44}$$

This allows us to write the 2D-3D registration problem as

$$\hat{\mathbf{q}} = \underset{\mathbf{q}\in\mathbb{R}^n}{\operatorname{argmax}} \sum_{\mathcal{M}} \overline{R}_S(T_{\mathrm{K}}(\mathcal{M}, \mathbf{q}, \mathrm{K_F})). \tag{8.45}$$

Solution of the Registration Problem. Please, compare the formalizations given above (equations 8.43 and 8.45) with equation 7.4 and notice, that we actually just modeled objective functions which can be solved via optimization. There is just the problem that we do not have a segmentation of the stent segment yet, of course, because it is actually our goal to find one. Thus, we have to formulate the objective functions such that they relate the parameters to estimate with either intensities or features in the given image.

[12] Frequently, such a notation using the letter T refers to a rigid transformation. Note that in our case, it actually deforms the model, because $\mathbf{p}$ also contains parameters affecting its shape.

8.3. Optimization

A general introduction to optimization has already been given in section 7.3. Thus, at this point, let us focus on the design of an objective function for the task at hand, which is to find some model parameters (defining pose and shape) such that the model matches best the image of a stent segment to find.

8.3.1. Objective Function Design

The objective function has to relate the model parameters with the information in the image somehow. Ideally, it maximizes at real stent segment positions and is minimal at other locations. Without any further preprocessing we have to rely on the only information that we get from the image, namely, the intensity values.

Intensity-Based Objective Function. From the analysis of the stent graft features in section 6.2 we know that the stent segments appear as bright contours in CT volumes. Given the fact that we can easily invert an image—and referring to section 8.4 where issues on preprocessing are addressed—we may also assume our interventional 2D images to show bright stent contours.

Hence, in order to keep it simple, we may try to use an objective function based on the brightness at model point locations; that is to say, if the model hits a lot of bright pixels or voxels we assume it to be at a good position and in "good shape". Formally, the objective function can be expressed by

$$f(\mathbf{p}) = \sum_{\mathcal{M}} I(T(\mathcal{M}, \mathbf{p})). \tag{8.46}$$

Now, it is the goal to maximize the objective function, of course:

$$\hat{\mathbf{p}} = \underset{\mathbf{p}\in\mathbb{R}^n}{\operatorname{argmax}} \sum_{\mathcal{M}} I(T(\mathcal{M}, \mathbf{p})) \tag{8.47}$$

with the input image I, as well as T and $\mathbf{p}$ as defined above.

Note that this formalization refers to 3D images. The 2D case is analogous, though: T just has to be substituted for T_{K}, and $\mathbf{p}$ for $\mathbf{q}$ (along with $\hat{\mathbf{p}}$ for $\hat{\mathbf{q}}$, respectively).

Distance-based Objective Function. The design of the objective function as done above (equation 8.46) has one serious drawback: It lacks the capability to guide the optimizer towards the desired result. Black pixels or voxels, for example, do not give the optimizer any valuable input on how far it is from the optimal solution. This shows that the objective function does not incorporate any information on the distance between the model and actual objects in the image. As a consequence, the optimizer lacks any input on where to go from the current estimation as compared to the last one. Thus, it is useful to model some distance information which can be used by the optimizer in order to improve its robustness and rate of convergence.

For this purpose we can use a *distance map* which contains the (approximate) distance of each pixel or voxel to the nearest object (i.e. pixels/voxels with intensities unequal to zero). Details on how such a map is built are given in section 8.4.1.

Let us assume we had such a distance map D. It gives us the ability to look up the distance of a model point to the nearest bright contour. Now, we may reformulate the objective function from equation 8.46 to utilize this information:

$$f_d(\mathbf{p}) = \sum_{\mathcal{M}} D(T(\mathcal{M}, \mathbf{p})). \tag{8.48}$$

In this case, we want to minimize the sum of distances given by the objective function, of course:

$$\hat{\mathbf{p}} = \underset{\mathbf{p}\in\mathbb{R}^n}{\operatorname{argmin}} \sum_{\mathcal{M}} D(T(\mathcal{M}, \mathbf{p})). \tag{8.49}$$

Again, the 2D case is analogous.

Using this formulation, the optimizer gets input from every point concerning its distance to the nearest object which allows it to alter the parameters such that the model is drawn to that object. At least, this approach works well for our 3D phantom data which is free from disturbing structures and noise. For real patient data, and especially for the 2D case, there are too many objects that clutter the distance map. Proper preprocessing like background removal and contour detection could help solve the problem. The former is addressed in the section 8.4.2.

Solution of the Optimization Problem. There is still the question left on how to solve the optimization problems formulated in the previous paragraphs. Note that we have not modeled any constraints so far. Furthermore, we do not have a linear relationship in the parameters to estimate. Hence, we are in the field of *unconstrained nonlinear* optimization. Various methods exist to solve this kind of problem. A few examples have been given in section 7.3.3. Here, we use the downhill simplex method.

8.3.2. Downhill Simplex Method

The *downhill simplex* method (also called *Nelder-Mead* method, or *amoeba* method) [NM65] is a common method for solving nonlinear optimization problems. It must not be confused with the likewise popular simplex method for solving problems in linear programming proposed by Dantzig. Nelder and Mead's method solves an unconstrained minimization problem in n dimensions iteratively utilizing a simplex. Its name comes from the fact that for each iteration the simplex is updated in a way that it "rolls downhill" in order to approach the minimum. No derivation of the objective function is necessary.

Principle. The method uses the concept of a simplex, which is a special polytope of $n + 1$ vertices in n dimensions. Examples of simplices are a line segment in 1D, a

triangle in 2D, a tetrahedron in 3D, etc. Every point corresponds to a set of parameters and its value can be computed by the objective function. The points are compared and the best and worst one are determined. Using a set of rules these are replaced. The basic three operations are *reflection*, *contraction* and *expansion*.

The simplest step is to replace the worst point with a point reflected through the centroid of the remaining n points. If this point is better than the best current point the simplex is expanded along this line. On the other hand, if this new point is not much better than the previous value we are stepping across a valley and, hence, the simplex is contracted towards a better point. The algorithm details can be found in [NM65].

Comparison to Other Methods. As opposed to the gradient-based methods which may be used as well in order to solve our optimization problem the downhill simplex method has the advantage that we do not need to compute gradients of the objective function. This means that we do not have to care about discontinuities of the latter. Furthermore, there is not need to estimate a proper step size. The drawback of slower convergence is outweighed by its simplicity and robustness. Problems like the convergence to an undesired local optimum or the premature contraction of one or several parameters to a constant value are common to all optimization algorithms.

Experiments showed that the downhill simplex method provides a rate of convergence high enough for our purpose. The low dimension of the parameter space allows for fast evaluations of the objective function and, hence, convergence time stays in reasonable limits. In addition, it is not very likely for other methods to be much more robust and, hence, we are doing fine so far with the chosen approach. A qualitative analysis and comparison of different optimizers is not subject to this thesis.

8.4. Preprocessing

Several preprocessing methods on the input images can help improve or facilitate the registration process. The full potential has not been tapped yet by this thesis, and it is up to future work to utilize more methods to improve the input images. However, some basic filtering operations have already been tried out and are summarized in this section.

8.4.1. Distance Transform

As already stated, an objective function based on distance information may increase the optimizer's robustness and rate of convergence. In order to get that information we can use a *distance transform* which can be based on different distance metrics. The distance transform generates a *distance map* containing the (approximate) distance of each pixel or voxel to the nearest object (i.e. pixels/voxels with intensities unequal to zero). Once the distance map has been built—which has to be done only once if the

image data is not changed—it allows for quick determination of the distance between the model at hand and the visible image structures.

Formally, the distance map is defined as

$$D(\mathbf{x}) = \min_{\mathbf{y} \in O} d_M(\mathbf{x}, \mathbf{y}). \tag{8.50}$$

with object O, background O' and distance measure M. Note that, although a distance transform is frequently applied to preprocessed binary images (using edge detection or contour extraction) this is generally not necessary. However, noise and undesired image structures may fill up the distance map and make it practically useless. Hence, some issues on background removal are addressed in section 8.4.2.

Distance Metrics. Due to the fact that is computationally exhaustive to compute *Euclidean distances* globally, several different distance metrics have been proposed in order to address this issue by reducing the *global* problem to involve only *local* filter operations (using filter masks). The objective is computational efficiency subject to a good approximation of the Euclidean distance. In [Bor84] some of the common distance metrics are compared. One of them is the *Chamfer distance*, which we want to examine at this point.

Chamfer Distance. The general idea is to assume that it is possible to deduce the distance value of a pixel by the distance values of its neighbors. Two different masks are applied to the values of the distance map in a forward and a backward scan:

Forward:

d_2	d_1	d_2
d_1	0	

Backward:

	0	d_1
d_2	d_1	d_2

The forward mask is moved over the image left to right and top to bottom, while the backward mask is moved in the opposite way. In each position, the sum of the local distance in each mask pixel and the value of the pixel it covers is computed, and the new value of the zero pixel is the minimum of these sums. In [Bor84], Borgefors suggests optimal values for the parameters d_1 and d_2 as well as practically useful ones. Using $d_1 = 3$ and $d_2 = 4$ (whereby the result has to be divided by 3) the result differs by a maximum of $\sqrt{2} - \frac{3}{4}$ from the Euclidean distance. This distance is referred to as Chamfer $(3, 4)$ distance.

Note that these masks apply to the 2D case. However, similar ones can be used for 3D images:

Forward:

d_3	d_2	d_3
d_2	d_1	d_2
d_3	d_2	d_3

and

d_2	d_3	d_2
d_1	0	

Backward:

	0	d_1
d_2	d_1	d_2

and

d_3	d_2	d_3
d_2	d_1	d_2
d_3	d_2	d_3

The forward mask is moved over the volume left to right, top to bottom, and front to back. Again, the backward mask is moved in the opposite way. According to [Bor84], good practical values for the parameters are $d_1 = 3$, $d_2 = 4$, and $d_3 = 5$ as integer approximation to the optimal ones. This distance is referred to as Chamfer $(3, 4, 5)$ distance and has an upper bound for the difference to the Euclidean distance of $\frac{\sqrt{7}}{3} - 1$.

Now we are able to use the Chamfer distance for the objective function defined by equation 8.48.

8.4.2. Background Removal

Using the objective function based on image intensities (equation 8.46) we are looking for bright curves in the image that form the stent segments. In order to compute a distance map, we also typically consider bright pixels or voxels to belong to objects. It has already been noted that in interventional 2D images these contours are usually dark. Although we may simply *invert* the image to get bright ones this is usually not sufficient. Due to the projectional nature of the images the latter are cluttered with different structures (like bones, catheters, implants, etc.) crossing the stent and making it hard to distinguish. In addition, image noise may be an issue. We can address these problems—at least partly—by applying *background removal* methods. One of them is the so-called *bothat filter*.

Bothat Filter. This technique is a combination of simple *morphological* and a *mathematical* operation: The original image is *subtracted* from a *closed* version (i.e. a *dilation* followed by a *erosion*) of the image. Closing removes islands and thin filaments of *background* pixels. After the subtraction only strong contours remain as bright features. The effect on our image data can be seen in figure 8.6. Experiments showed that this kind of preprocessing is a first step in the right direction.

8.4.3. Further Methods

The potential of preprocessing methods has not been tapped by far. This is definitely an issue for future work as there is still a lot of room to improve the input images for the registration procedure. Most of all, a proper *edge* or *contour detection* could help the registration consider only dominant and important contours or serve as input for the generation of a distance map. Some further ideas and first experiments are outlined in section 10.2.1.

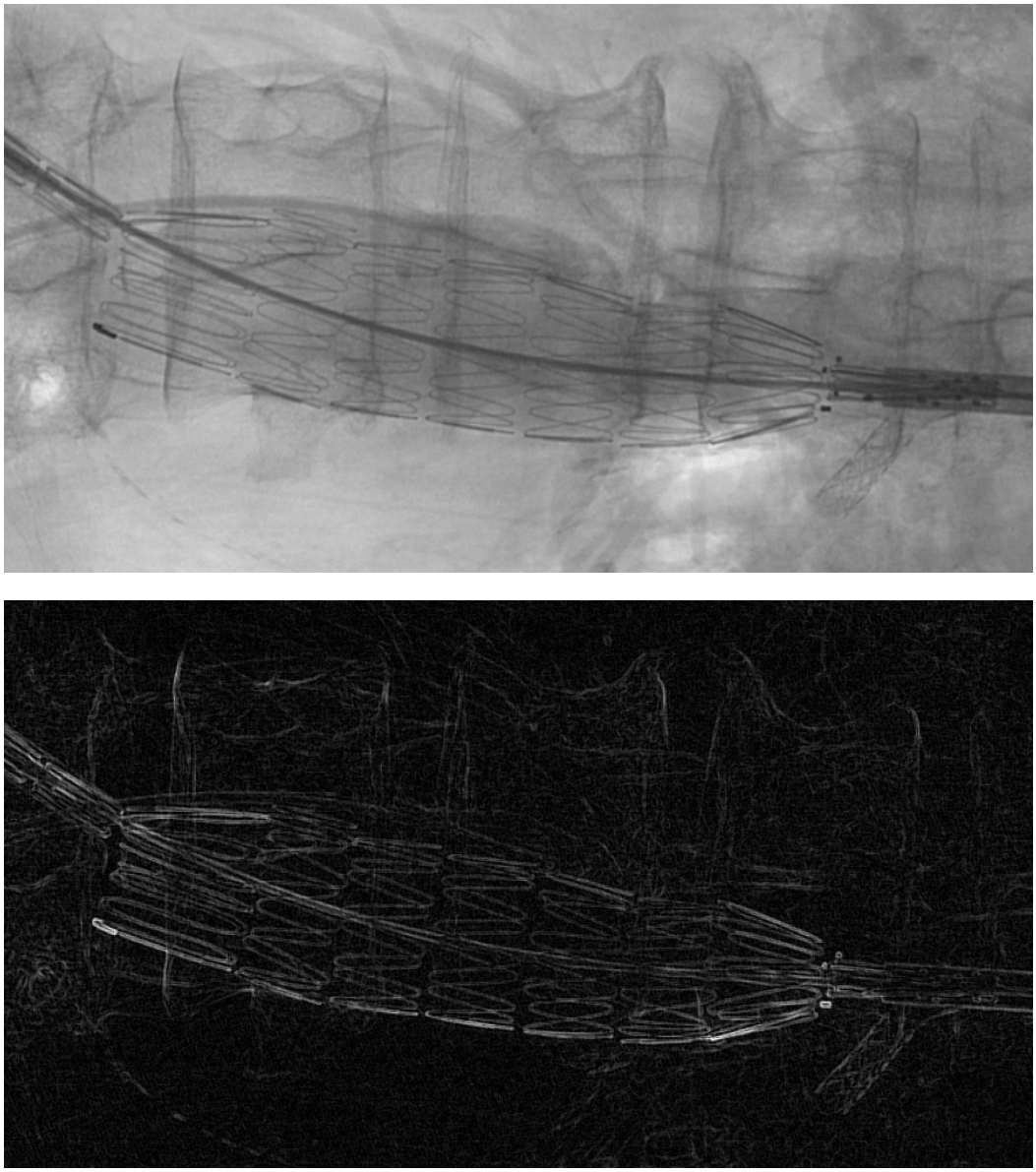


Figure 8.6.: Original angiogram with stent graft (top) as opposed to the bothat-filtered version (bottom). The gray background is removed along with some noise and unwanted structures, and the stent segments . (Note that the filtered image has been contrast stretched for better visibility, which adds back some noise.)

Part IV.

Results and Conclusion

9. Experimental Results

After the description of the developed methods and models it is time to test the approach on phantom data as well as real patient data. Starting from a general analysis of important evaluation criteria for the registration process the experimental setup is described along with the used test data, the collection of ground truth and the tested parameters. Furthermore, the experimental results are evaluated and discussed.

9.1. Evaluation Criteria

According to [PC08] five criteria should be addressed when evaluating a registration algorithm. These are *execution time*, *accuracy* in the region of interest, *breakdown point*, *automation*, and *reliability*. Although the chapter in the mentioned book refers to (rigid) image registration, the defined criteria are still viable for model-to-image registration.

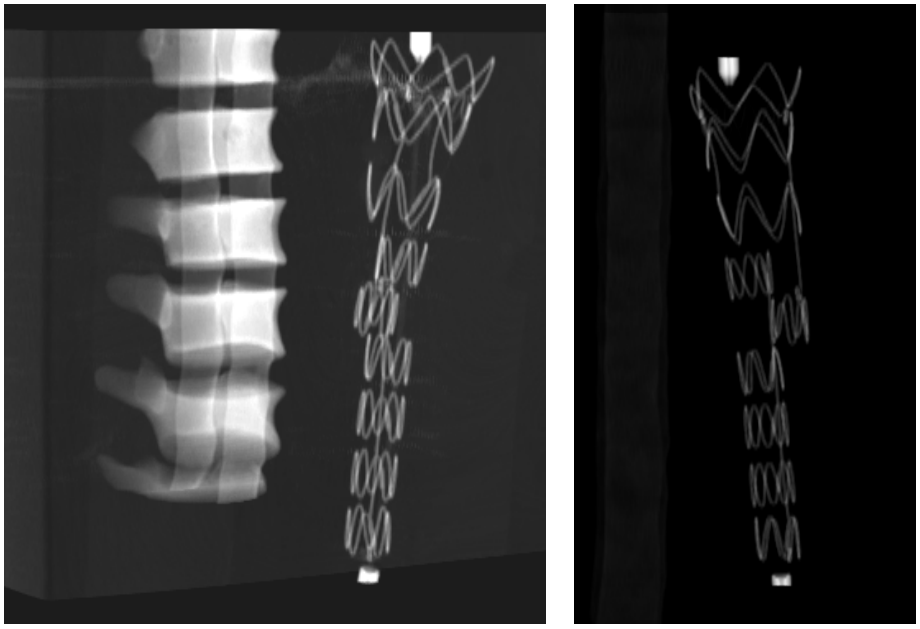
As already stated, we use a semi-automatic approach. Due to the lack of experts' segmentations the issue of accuracy has not been addressed yet—just as an analysis of the breakdown point. However, some first experiments give hints on the reliability and the execution time dependent on the initialization.

Time permitted an in-depth examination of all these parameters. However, we want to look at some examples, and address possibilities for a qualitative analysis. We want to focus on the initialization of the individual segments as key to a proper registration process. Furthermore, the experiments were conducted mainly with 2D phantom and patient data since this is more challenging than the 3D case, anyway, due to missing 3D information leading to overlapping structures.

9.2. Experimental Setup

First, we want to take a look at the image data used for the tests, along with the general testing procedure, including the gathering of ground truth and the error measurement.

Test Data and General Procedure. For the phantom data a stent graft has been attached to a box with a spine phantom, and scanned by a CT scanner with high resolution in order to get high-quality images (see figure 9.1(a)). *Digitally reconstructed roentgen (DRR)* has been used in order to produce 2D images similar to radiography. The projections have been chosen in a way that the stent graft is not overlaid with the spine phantom such that tests on noise-free "perfect" data can be run.



(a) CT scan of 3D phantom. (b) DRR of phantom.

Figure 9.1.: CT scan (a) and DRR (b) of the phantom: The CT image has been acquired with high resolution for best quality. The projection has been chosen in a way that the stent graft is not overlaid with the spine phantom.

Some interventional 2D images (fluoroscopy/angiograms) serve as real patient data for testing purposes (see figure 8.6).

The registration is tested via the developed "VirtualStenting" application (see section A.1). Parameters are altered systematically within certain bounds and results are compared to ground truth.

Ground Truth and Error Measurement. Commonly, comparison to experts' segmentations are made. We are lacking these so far. Hence, we performed a manual initialization on each segment by putting the model as close as possible to the actual segment and registering it finally to get a (more or less) exact match.

The error of the test registrations compared to this ground truth segment model $\mathcal{T}$ is measured by the Euclidean distance, i.e. the sum of the Euclidean distances of corresponding model points normalized by the number of model points:

$$d_E(\mathcal{T}, \mathcal{M}) = \|\mathcal{T}, \mathcal{M}\|_2 = \frac{1}{n} \sum_{i=0}^{n} \|\mathbf{t}_i - \mathbf{m}_i\|_2 \tag{9.1}$$

(a) Semi-automatic registration of leg component. (b) Expansion. (c) Shrinkage.

Figure 9.2.: Registration on 2D phantom data: (a) Starting from an initialization via bounding box the four segments of a stent graft leg component are detected. (b) Bad initialization may result in expansion or shrinkage of the model.

with $\mathbf{t}_i \in \mathcal{T}$, $\mathbf{m}_i \in \mathcal{M}$, and n the number of points[1].

9.3. Experiments on Image Data

Experiments on the image data include 2D phantom and real patient data. Qualitative analysis is just about to start and, hence, a lot of issues are up to future work. However, first experiment give insight into the capabilities and weaknesses of the proposed approach.

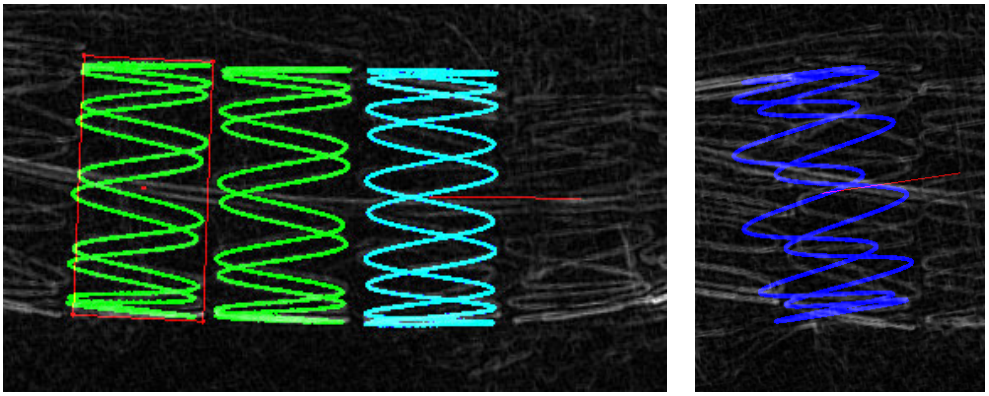
9.3.1. Phantom Data

First experiments on the 3D phantom data yielded good and robust results due to the lack of noise. A qualitative analysis has not been made so far, though. A first step towards that direction has been undertaken by the 2D phantom images. Figure 9.4 illustrates the registration's dependency on the initialization of the pose parameters (translation and rotation). Translation has been applied to the pose of the ground truth model for up to two times the amplitude or radius of the segment. The effect is tested on each direction individually. Likewise, rotations are altered for up to 45 degree.

Figure 9.4(f) shows the special case where the segments are initialized with the standard shape (i.e. all rotations are zero except for the rotation about the image plane). It may be interesting to examine this issue, because our approach of iterative registration is based on this initialization. One can see that the results are not that accurate anymore. We may reconsider the strategy and try to initialize all rotations according to the previous segment. Tests and comparisons will follow.

On the other hand, figure 9.5 illustrated the effect of altering amplitude and radius (i.e. both radii in the same way). Finally, figure 9.2 shows cases for a good semi-

[1]Note that for comparison, the models have to be sampled in the same way in order get corresponding points. The ground truth determines the sampling period



(a) Good iterative registration of 3 segments. (b) Problematic case.

Figure 9.3.: Registration on patient data: (a) The three segments have been registered automatically after initialization of the first segment. The last one is not accurate, though. (b) A problematic case: The model is drawn to the contour given by the catheter.

automatic registration along with situations that can occur if the initialization is too far away from the real object. These includes shrinkage or expansion of the model.

Without going too much into detail, we can say that the registration in general works well given some reasonable bounds on initialization. However, one can see that the cost function does not necessarily have a dominant maximum for good registration results, and that it has a lot of outliers. A re-design of the objective function may improve that issue.

9.3.2. Real Patient Data

Patient data has not been addressed in a qualitative way, yet. First subjective impressions yield a lesser robustness due to a lot of image noise. However, given some good initialization, the results are not too bad, and it is promising that improvements to preprocessing as well as the objective function can increase robustness.

Figure 9.3 illustrates examples of good results as opposed to some open challenges. These are mainly related to strong contours such as given by the catheter.

9.4. Summary

The results of the approach presented in this thesis are promising. The process of qualitative evaluation is just about to start, though. Further tests should contain analysis of the effect of image noise (e.b. by adding noise to the phantom data) and are supposed to include phantom and image data in 2D and 3D. Furthermore, experts'

segmentations have to be acquired in order to compare the results of the registration in terms of accuracy.

Finally, a couple of improvements could increase robustness and accuracy. A couple of ideas are given in the following chapter. After the incorporation of some of them the testing process will be resumed and detailed.

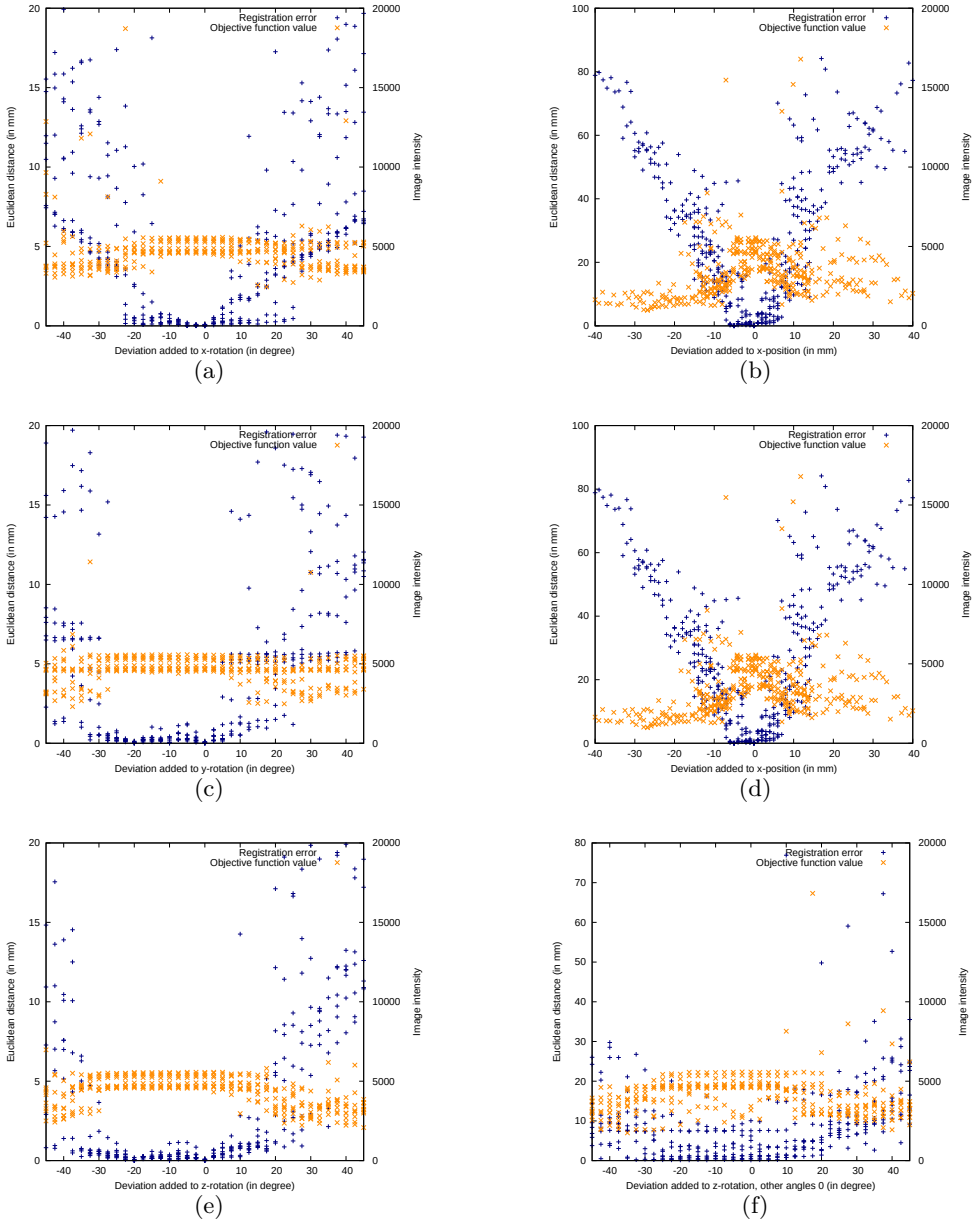


Figure 9.4.: Registration error and objective function dependent on pose initialization.

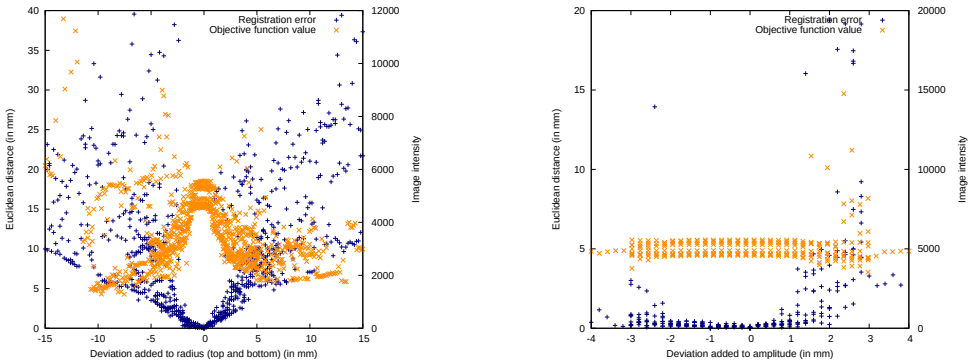


Figure 9.5.: Registration error and objective function dependent on shape initialization.

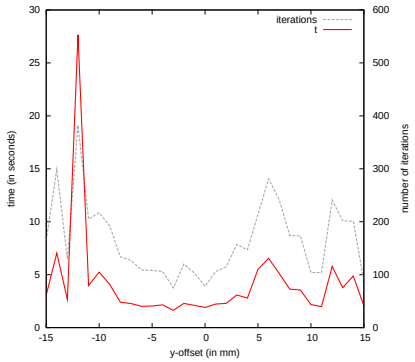


Figure 9.6.: Dependency of execution time and iterations on the initialization (exemplary for one segment and y-offset).

10. Conclusion and Outlook

After the evaluation of the experimental results, this chapter discusses the reached goals and open challenges. Furthermore, it gives an outlook on future work by providing some ideas on improvements to and possible extensions to the developed approach. The latter include advanced preprocessing methods, redesign of the objective function, segmentation-driven registration, and the utilization of more a priori knowledge for constraints on possible model locations. Ideas on full automatization of the registration process conclude this chapter. Finally, the closure completes this thesis.

10.1. Discussion

The experiments on phantom data as well as real patient data showed that the developed approach works in general, given reasonable prerequisites and initializations. They also revealed some issues to be addressed in order to improve the approach. Let us summarize the reached goals and the open challenges.

10.1.1. Reached Goals

As the title states, the general objective of this thesis was to develop an automatic method for the detection of stent grafts in 2D and 3D image data. More specifically, an approach based on segmentation via registration was supposed to be evaluated. The choice for this way has been adequately constituted for, and the required methods and model have been developed, implemented and evaluated.

We could show that the approach is a step in the right direction and works under reasonable prerequisites. The latter are approximate measurements of the stent segments' dimensions and distances in the stent graft along with the initialization of the first segment's pose. The former can easily be obtained prior to the application (intervention or postoperative control and monitoring) and have to be gathered only once per stent graft in use. The latter is dependent on the image dimension: 3D pose initialization is a little bit more cumbersome, but is not up to hard time constraints. However, it should take less than a minute for a trained person (including the registration), which is surpassingly small compared to a manual segmentation process. 2D pose initialization, on the other hand, is typically required during the intervention, and can be accomplished by just four mouse clicks, which can easily performed intraoperatively.

On the whole, the developed approach provides a remarkable speed-up as opposed to manual segmentation, accompanied by more objective and reproducible result. There are definitely a couple of open challenges, though, but this thesis may provide a good

starting point for fuhrer research in the direction. In the future, clinics, physicians as well as patients may profit from these benefits.

10.1.2. Open Challenges

Although the approach presented here yields more reproducible results than manual segmentation, it is still dependent on the pose initialization of the first stent segment. This implies that the results may not be totally reproducible, and that the goal of full automatization is not reached yet. However, some suggestions on addressing this issue are given in section 10.2.5.

Especially in the case of real patient 2D image data, the approach is not robust enough yet for the clinical routine. Open challenges include proper image preprocessing and improvements on the objective function used for optimization. One problem is the fact that the stent segment model gets drawn to unwanted structures easily. Especially catheters pose a challenge. Some ideas on solving these issues are presented in section 10.2.

Another important issue must not be concealed either: Although stent segment offsets from the main axis have been incorporated into the XML model specification (see section A.2) it is an open issue how to deal with them. Since they could be rotated around the stent graft's axis the pose initialization becomes a challenge. Section 10.2.4 gives ideas to address this problem.

After the incorporation of some improvements, additional evaluations of the approach on a larger data set including comparison to experts' segmentations are indispensable.

10.2. Ideas and Future Work

Referring to the open challenges in the previous section, there are some issues to be addressed in order to improve or extend the presented approach. These include more advanced preprocessing methods, redesign of the objective function, the incorporation and combination of further methods, and utilization of more a priori knowledge.

10.2.1. Advanced Preprocessing Methods

Given the generally noisy real patient data, more advanced preprocessing methods can improve the input images for the registration process, possibly yielding more accurate results and robustness. The following list gives a couple of suggestions:

Noise suppression can reduce image noise. Smoothing operations are typically used for the task. Methods reach from simple linear techniques like *mean* and *Gaussian filtering* to more advanced ones, such as *median* and *anisotropic diffusion filtering*.

Contrast enhancement could improve the visibility of stent segments in the images. In order to keep image noise low while additionally enhancing the desired structures, *shape-preserving* methods may be chosen.

Edge detection could filter out relevant edges and contours for the registration process. Common techniques are the *Sobel filter*, *Laplacian filter masks* and the *Canny edge detector*. First experiments with an approach based on *steerable filters* [FA91] showed that it may contain valuable information on local directions of contours. The *Frangi filter* (frequently used for vessel segmentation) might also yield good results due to its capabilities of analyzing tubular structures.

10.2.2. Objective Function Redesign

The shortcomings of an objective function purely based on image intensities have already been explained in section 8.3.1. The approach based on distance maps works well for our 3D phantom data which is virtually free from disturbing structures and noise. It is not applicable to real patient data (especially for the 2D case), since there are too many objects that clutter the distance map. Advanced preprocessing as outlined in the last section could help solve this issue by means of edge detection. Extracted contours might facilitate the generation of a more useful distance map, or allow for a complete redesign of the objective function in order to make use of the additional information.

Referring to the problem that the stent segment model gets drawn to other structures such as catheters too much, an analysis of the present contours in the image could help sort out the features in question.

A further idea was to introduce weights giving segment peaks a higher importance. That way, straighter parts of the segment model contribute less to the objective function value, which constrains the influence of long structures such as catheters, and lets the registration process regard more peak-like structures. First experiments and a subjective evaluation yielded better performance for segments near catheters, but, on the other hand, introduced other problems.

Regarding the very simple designs of the objective functions utilized so far, the results are not too bad—and there is still a lot of room left for improvement. A more elaborate objective function, in conjunction with advanced preprocessing, may yield more accurate results and enhance the robustness of the registration process.

10.2.3. Segmentation-Driven Registration

In [GBHN07], Groher *et al.* presented an approach that combines segmentation and registration in an iterative process for vessel extraction. The results of segmentation and registration influence each other combining the strengths of both methods providing a robust automatic feature-based registration with high capture range. The algorithm is very promising and yielded good results for their purpose. However, time permitted a further investigation of its applicability for the approach presented here and, thus, it is up to future work.

10.2.4. Utilizing More A Priori Knowledge

Referring to section 6.2 one might realize that we have not tapped the full potential of our a priori knowledge yet. We could use some of the given information in order to reduce the search space for stent segments.

Vessel Segmentation and Stent Pose. Given the fact that the stent graft is always situated within the aorta—well, except something *really* serious had happened—we might be able to incorporate knowledge from a prior *vessel segmentation* in order to restrict the detection process to a smaller region. In addition, the aorta centerline could impose constraints on segment orientations. Of course, the vessel segmentation is a prerequisite in this case, too, but we can assume that it usually is to be performed anyway in order to examine the aneurysm (like measuring its size, etc.). This is especially the case for post-operative monitoring. Intraoperatively, however, we would either have to perform the vessel segmentation on the fly or register a vessel model generated from a segmentation that is usually carried out preoperatively for path-planning purposes. Deformations of the vessel that occur during the intervention might make this impracticable or at least very complex, though. Certainly, some further research on this topic would shed more light on that issue.

Constraints on Spacial Relationships Between Segments. Since the individual stent segments are attached to the graft, they cannot occupy arbitrary locations in the image. Hence, a further way to reduce the amount of possible pose configurations of stent segments would be to introduce constraints on their poses regarding spacial relationships between them. The latter could depend on physical properties of the stent graft considering maximum amounts of torsion and kinking. We could perform registration on a whole constrained stent model, i.e. all segments at once, or iteratively but with constraints on distances and rotations. Stacking up the segments into a complete stent model could also facilitate to deal with segment offsets from the center axis by letting the registration process rotate the complete model around. An iterative approach alternating between registration of the whole model and a refinement of the individual segments is also imaginable.

10.2.5. Towards Full Automatization

Based on the upper section we may be able to address the aspects of a fully automatic detection approach. The reduction of the search space to look for stent segments may also reduce computational costs for exhaustive feature search. The latter is necessary to automatically estimate possible stent segment locations in the image in order to initialize the registration procedure. Methods that could be utilized for this purpose include *template matching* and *Hough transform*.

Template Matching. This method is common in computer vision, and requires the generation of templates that represent the objects of interest. These templates are moved and rotated across the image and comparison of intensity values or distances to features determine the quality of the match at a certain location.

Taking the stent segment model as template and allowing for arbitrary rotations in three dimensions would certainly be too costly. However, making use of the prior knowledge presented in the last section (10.2.4) it may be possible to reduce the search space and apply restrictions on rotations. In 2D, it might be sufficient to use the standard configuration, that is to say, all rotations are set to zero except for the rotation about the image plane normal. A couple of different templates, or a composition of a template from several configurations is also imaginable.

Once a couple of good candidates have been found, further analysis on their spatial relations may help detect outliers and search for further segments.

Hough Transform. This method is commonly used to detect lines in images. It may be extended to search for circles or any other feature that can be modeled parametrically[1]. The Hough transform is performed on a gradient or feature image and transforms every point from Cartesian coordinates into the parameter space where it represents a curve. Points belonging to the same feature result in intersecting curves in the parameter space. Clustering algorithms allow for detection of these intersections and transforming the clusters back into the original coordinates yields the image features.

Since we already have a parametric model of a stent segment it may be used by the Hough transform in order to detect feasible segment candidates. There is one major drawback: Although there are efforts to improve the method's efficiency, it is computationally intensive. The costs increase with the parameter space. The complexity is $\mathcal{O}\left(D^{n-2}\right)$ where D is the image space and n is the number of parameters. Hence, it is rarely used for features other that lines ($n = 2$) and circles ($n = 3$). However, the stent segment model is in the dimensions of $n = 8$ (2D) or $n = 9$ (3D).

It is up to future word to test the method on our problem. Maybe, a reduction of the search space using a priori knowledge is enough to reduce the computational costs to a reasonable level. A further idea is to consider segment peaks only, which resemble parabolas. The latter can be parametrized by a single parameter (via $x = t, y = t^2$) resulting in a 1D Hough space which can be examined at low costs. However, this implies further (probably exhaustive) analysis of the derives parabolas concerning their spatial relations in order to derive stent segment candidates.

An approach based on the Hough transform may be worth a try in order to address the challenge of a fully automatic stent graft detection with the benefits of completely reproducible results.

[1] A modified version called *generalized* Hough transform allows for utilizing features that are ineligible for parametric modeling.

10.3. Closure

While a couple of implementation details concerning the developed application are given in the appendix, this section including discussion, outlook and ideas actually concludes the thesis. May it be a useful resource for further development in this direction such that the developed method becomes a basis for various applications, and a valuable tool for the clinical routine—making things a bit easier for physicians and personnel working in the related field.

Appendix

A. Implementation Details

The first section of this chapter contains some details on the “VirtualStenting” application that has been implemented in order to test the developed approach. Its usage is explained and illustrated by screenshots. A further section is dedicated to the model specification which describes the XML file that is used for storing stent graft properties which are required for the registration process. The template file is included in order to exemplify its usage.

A.1. VirtualStenting Application

The following sections are included from the automatically generated Doxygen documentation and describe the VirtualStenting application that has been created in order to develop and test the presented methods. It consists of a simple GUI that allows the user to load 2D and 3D image data as well as stent models and to control model and registration properties. Details on the usage are explained in the following sections and the model specification can be found in section A.2. Figure A.1 shows a screenshot of the application.

A.1.1. Introduction

The VirtualStenting application detects AAA stents in 2D and 3D images (e.g. CT(A), fluoroscopy, angiogram) (semi-)automatically. This is done by registering a stent model to the image data.

A.1.2. Usage

After the main window pops up, open an image via the menu ("File → Open") using the file dialog. The menu entry "Tools → VirtualStenting" opens a 3D viewer - also for 2D images (for model visualization purposes). The functionality is just taken from the "CAMP Image Viewer". Use left mouse button to rotate, right button to move image and middle button to zoom and rotate around view axis.

The "Stent Registration" dialog can be accessed from the corresponding button and controls the first stent segment model (appearance and pose) as well as registration and optimization properties (tolerances, parameters to optimize including step sizes). There, you can play around with registering a single segment.

On start-up, a complete model is loaded from the template file (see class StentSegmentModel) which can be changed via the model file dialog by pressing [l]. Verify

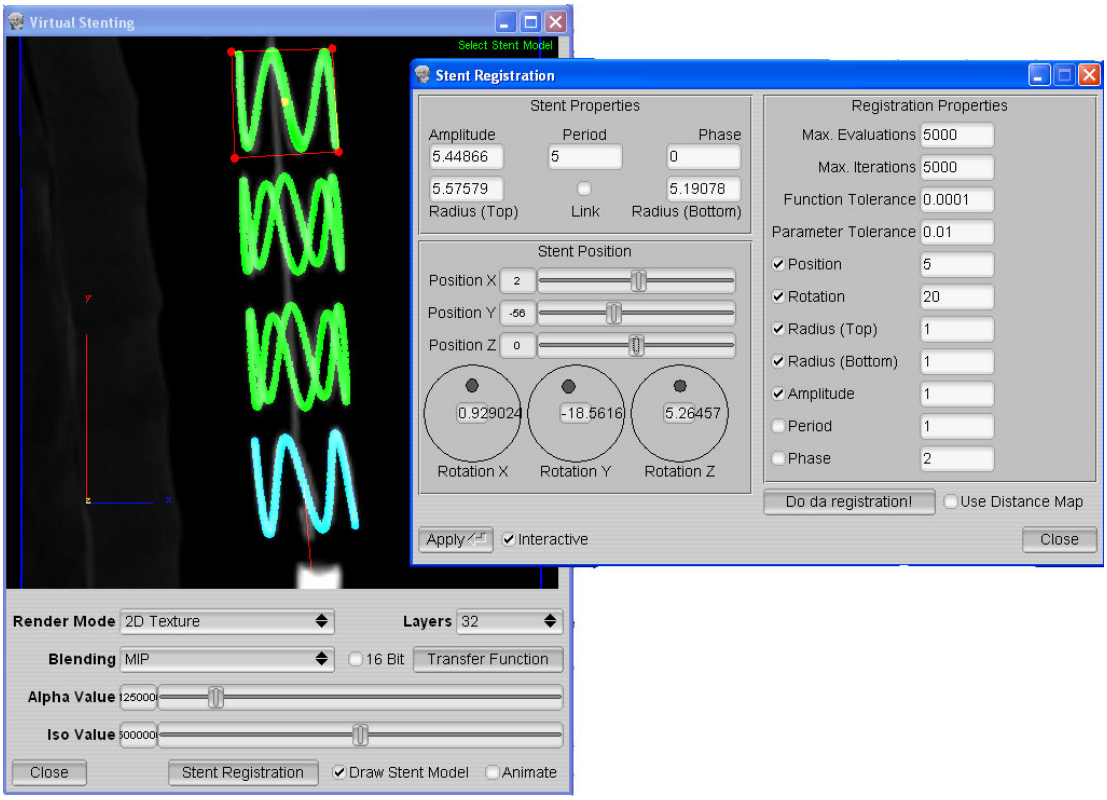


Figure A.1.: VirtualStenting application: The "Stent Registration" dialog allows for the adjustment of stent segment properties, and provides controls for position and rotation. The right hand side of the dialog contains registration options, such as the maximum number of objective function evaluations and iterations as well as function and parameter tolerances. Furthermore, the parameters to estimate can be specified along with corresponding optimizer step sizes. The loaded image shows the leg component of a stent graft derived from a CT scan and projected to 2D via DRR. The registered model is overlaid on top of it. In 2D, the first (= top) segment can be initialized not only by using the pose controls, but also via a bounding box specified by four mouse clicks. After successful registration of the first segment, the other ones are initialized and detected automatically.

that the correct stent model is loaded. (The first segment can be visualized/hidden by pressing [1], the full model by pressing [2]. Key [3] toggles between 2D and 3D model. See section "Shortcuts" for all options.)

In case of a 3D image, the first segment has to be moved to the initial position manually via the corresponding controls in the "Stent Registration" dialog. In case of a 2D image - and after pressing [s] - you can select the first (= top) segment by defining its bounding box: Just click on the segment's corners starting with the "upper left" one. The right mouse button undoes the last click. As soon as four points are selected registration starts automatically. (Press [v] to see what's going on during optimization.) Once the first segment is registered, press [r] to register the remaining segments. The middle mouse button deletes the bounding box and resets the model.

A.1.3. Notes

The optimizer maximizes image intensities. Thus, use images that show the stent in a light color with dark background. (Invert the image if necessary.) For 2D images preprocessing is required. You may try to use bothat-filtered images. Additional processing might improve the results.

A.2. XML Model Specification

The stent model can be specified and loaded into the "VirtualStenting" application via an XML file. In order to exemplify its usage the template file that serves as a basis for building stent graft models is included here:

```
<?xml version="1.0" encoding="utf-8" ?>

<!-- Stent model template file for AAA stent grafts

File:   stent_AAA_model_template.xml
Author: Markus Urban
Date:   2009-06-06

Purpose:
  This XML file serves as AAA stent model specification that can
  be used in the "VirtualStenting" application for (semi-)automatic
  stent detection in 2D and 3D image data.

Notes on building the stent model:
  The stent model consists of various segments. Within the model,
  you can choose default values (mandatory) for various segment
  properties, like e.g. radius and height. If one or several segments
  deviate from the defaults or have a center offset you can specify
  these values within the <segment> tag with the according segment
  number (from top to bottom). If you do not provide a number the
  segments are numbered automatically starting with 1 or the first
  number given. That way, you can compose a complete model with
  different segments. All values are optional (defaults are used where
```

```
  values are not specified).
-->

<!-- Stent model template: -->
<stent-model>

  <!-- These are the defaults for all segments;
       tags should contain values that correspond to most/all of the
           segments; all tags are mandatory: -->
  <segment-radius>5</segment-radius>
  <segment-height>5</segment-height>
  <segment-peaks>5</segment-peaks>
  <inter-segment-distance>15</inter-segment-distance>
  <number-of-segments>8</number-of-segments>

  <!-- Optionally specify segments that deviate from defaults like this;
       # stands for the segment number from 1 (top segment) to
           <number-of-segments> (bottom segment); all tags are optional:
  <segment no="#">
    <radius></radius>
    <height></height>
    <peaks></peaks>
    <distance-to-next></distance-to-next>
    <center-offset></center-offset>
  </segment>
  -->

  <!-- Segment examples: -->
  <segment no="1"> <!-- "no" is optional here (starts from 1 anyway) -->
    <radius>10</radius>
    <height>8</height>
    <distance-to-next>25</distance-to-next>
  </segment>
  <segment> <!-- automatically gets no="2" -->
    <radius>8</radius>
    <height>6</height>
    <distance-to-next>20</distance-to-next>
  </segment>
  <segment no="4"> <!-- segment 3 gets defaults and 4 deviates again -->
    <center-offset>10</center-offset>
  </segment>

</stent-model>
```

References

[Bor84] G. Borgefors, *Distance Transformations in Arbitrary Dimensions*, Computer Vision, Graphics, and Image Processing, 27 (1984), no. 3, pp. 321–345. 70, 71

[ČAW+06] P. Čech, A. Andronache, L. Wang, G. Székely, and P. Cattin, *Piecewise Rigid Multimodal Spine Registration*, in Bildverarbeitung für die Medizin 2006, no. 3 in Informatik aktuell, Springer Berlin Heidelberg, May 26, 2006, pp. 211–215. 32, 42

[Coo] Cook Medical, *Cook Medical | Aortic Intervention*. `http://www.cookmedical.com/ai/home.do`.

[CT04] T. F. Cootes and C. J. Taylor, *Statistical Models of Appearance for Computer Vision*, Mar. 2004. 41

[dBNMV03] M. de Bruijne, W. Niessen, J. Maintz, and M. Viergever, *Localization and Segmentation of Aortic Endografts Using Marker Detection*, Medical Imaging, IEEE Transactions on, 22 (2003), no. 4, pp. 473–482. 31

[FA91] W. T. Freeman and E. H. Adelson, *The Design and Use of Steerable Filters*, Pattern Analysis and Machine Intelligence, IEEE Transactions on, 13 (1991), no. 9, pp. 891–906. 85

[GBHN07] M. Groher, F. Bender, R. Hoffmann, and N. Navab, *Segmentation-Driven 2D-3D Registration for Abdominal Catheter Interventions*, in Medical Image Computing and Computer-Assisted Intervention – MICCAI 2007, vol. 4792 of Lecture Notes in Computer Science, Oct. 2007, pp. 527–535. 85

[GMW82] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*, Academic Press, Jan 1982. 52

[GOR] GORE Medical Products, *GORE EXCLUDER ® AAA Endoprosthesis*. `http://www.goremedical.com/excluder`.

[Gri01] C. Grimm, *Einführung in die Systemtheorie für Informatiker - Skript zur Vorlesung im WS 2001*. `http://www.ti.informatik.uni-frankfurt.de/grimm/skript/skript.html#skriptch4.html`, Sept. 2001.

[HZ04] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2nd ed., Mar. 2004. 63

[Jäh05] B. Jähne, *Digital Image Processing*, Springer, Berlin, 6th ed., June 2005. 38

[KRO+08] A. Klein, W. K. Renema, L. J. Oostveen, L. J. S. Kool, C. H. Slump, X. P. Hu, and A. V. Clough, *A Segmentation Method for Stentgrafts in the Abdominal Aorta from ECG-Gated CTA Data*, in Medical Imaging 2008: Physiology, Function, and Structure from Medical Images, vol. 6916, San Diego, CA, USA, Mar. 2008, SPIE, p. 69160R. 31, 32

[LRF08] I. Larrabide, A. Radaelli, and A. Frangi, *Fast Virtual Stenting with Deformable Meshes: Application to Intracranial Aneurysms*, in Medical Image Computing and Computer-Assisted Intervention – MICCAI 2008, vol. 5242 of Lecture Notes in Computer Science, 2008, pp. 790–797. 31

[Med] Medtronic, *Stent Grafts (Aortic Aneurysms) – Medtronic*. `http://www.medtronic.com/our-therapies/stent-grafts/index.htm`.

[MSN+06] J. Mattes, I. Steingruber, M. Netzer, K. Fritscher, H. Kopf, W. Jaschke, R. Schubert, J. M. Reinhardt, and J. P. W. Pluim, *Quantification of the Migration and Deformation of Abdominal Aortic Aneurysm Stent Grafts*, in Medical Imaging 2006: Image Processing, vol. 6144, San Diego, CA, USA, Mar. 2006, SPIE, pp. 61440V–9. 30

[NM65] J. A. Nelder and R. Mead, *A Simplex Method for Function Minimization*, The Computer Journal, 7 (1965), no. 4, pp. 308–313. 68, 69

[PC08] T. Peters and K. Cleary, eds., *Image-Guided Interventions: Technology and Applications*, Springer, 1st ed., May 2008. 8, 10, 11, 13, 37, 40, 75

[PH04] M. Pharr and G. Humphreys, *Physically Based Rendering: From Theory to Implementation*, The Morgan Kaufmann Series in Interactive 3D Technology, Morgan Kaufmann, Aug. 2004. Chapter 7, *Sampling and Reconstruction*, available online as free preview at `http://graphics.stanford.edu/~mmp/chapters/pbrt_chapter7.pdf`. 59

[PVB+04] M. Prinssen, E. L. G. Verhoeven, J. Buth, P. W. M. Cuypers, M. R. H. M. van Sambeek, R. Balm, E. Buskens, D. E. Grobbee, and J. D. Blankensteijn, *A randomized trial comparing conventional*

and endovascular repair of abdominal aortic aneurysms, The New England Journal of Medicine, 351 (2004), no. 16, pp. 1607–1618. PMID: 15483279. 20

[PXP00] D. L. Pham, C. Xu, and J. L. Prince, *Current Methods in Medical Image Segmentation*, Annual Review of Biomedical Engineering, 2 (2000), pp. 315–337. PMID: 11701515. 37

[QSS06] A. Quarteroni, R. Sacco, and F. Saleri, *Numerical Mathematics*, Springer, 2nd ed., Nov. 2006. 52

[VvSH+07] J. J. Visser, M. R. H. M. van Sambeek, T. H. Hamza, M. G. M. Hunink, and J. L. Bosch, *Ruptured Abdominal Aortic Aneurysms: Endovascular Repair versus Open Surgery – Systematic Review*, Radiology, 245 (2007), no. 1, pp. 122–129. 20

List of Figures

List of Tables