TECHNISCHE UNIVERSITÄT MÜNCHEN
Computer-Aided Medical Procedures & Augmented Reality

Dynamic Variational Level Sets for Cardiac 4D Reconstruction

Andreas Keil

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr. Daniel Cremers
Prüfer der Dissertation:	1. Univ.-Prof. Nassir Navab, Ph. D. 2. Prof. Rebecca Fahrig, Ph. D., Stanford University, U.S.A.

Die Dissertation wurde am 25. Mai 2010 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 2. September 2010 angenommen.

This work was partially funded by Siemens AG, Forchheim, Germany.

It is licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License. To view a copy of this license, visit `http://creativecommons.org/licenses/by-sa/3.0/` or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

Abstract

An increasing number of cardio-vascular procedures combined with the trend toward less invasiveness of these interventions demand intra-interventional imaging using flexible C-arm systems. Although high speed, dual source, 4D CT angiography recently emerged as a diagnostic imaging modality with a high negative predictive value, the ultimate goal for cardio-vascular imaging is a 3D visualization of the intra-operative situation, combining diagnosis, navigation, and quantitative evaluation.

The reconstruction of the coronary vasculature from angiographic C-arm sequences, however, is an order of magnitude more complex than that of other anatomy due to the inevitable motion blur associated with these acquisitions and the small size of the structures of interest. A successful and dose-efficient reconstruction algorithm therefore has to compute the cardiac motion in addition to the structural information.

Based on that application, this thesis presents a novel method for purely image-based 4D shape reconstruction without prior data. Adapting and extending mathematical models that originated from other disciplines, new methods for the reconstruction of dynamic shapes from arbitrary projections are developed. The main contributions are the introduction of dynamic level sets, combining implicit shapes with explicit motion information, and the probabilistic derivation of a new kind of energy functional for symbolic shape reconstruction. Experiments ranging from synthetic to phantom to real data sets prove the feasibility and versatility of the approach.

Keywords
Cardiac cone beam CT, Dynamic shape reconstruction, Level set methods, Active contours

Zusammenfassung

Die zunehmende Zahl von kardiovaskulären Eingriffen sowie der Trend zu minimalinvasiven Prozeduren verlangen nach intra-interventionellen bildgebenden Verfahren unter Verwendungen von C-Bögen. Obwohl mittlerweile schnelle, 4D-CT-Angiographie-Systeme mit zwei Röntgenquellen zur diagnostischen Bildgebung verfügbar sind und dieses Verfahren einen hohen negativen Vorhersagewert aufweist, bleibt eine 3D-Visualisierung der intra-operativen Situation, die zugleich Diagnose, Navigation und quantitative Evaluierung erlaubt, das Ziel.

Die Rekonstuktion der Herzgefäße aus angiographischen C-Bogen-Sequenzen ist jedoch durch die unvermeidbare Bewegungsunschärfe dieser Aufnahmen sowie die Größe der relevanten Strukturen eine ganze Größenordnung schwieriger als die anderer Anatomien. Ein erfolgreicher und dosiseffizienter Rekonstruktionsalgorithmus muss daher zusätzlich zur strukturellen Information auch die Herzbewegung berechnen.

Von dieser Anwendung ausgehend, präsentiert diese Arbeit eine neue Methode zur rein bild-basierten 4D-Formrekonstruktion ohne Vorwissen. Durch Anpassung und Erweiterung mathematischer Methoden, die anderen Disziplinen entstammen, werden neue Methoden zur Rekonstruktionen von dynamischen Strukturen aus beliebigen Projektionen entwickelt. Die Hauptbeiträge dieser Arbeit sind die Einführung von dynamischen Niveaumengen, zusammengesetzt aus impliziten Oberflächen und expliziten Bewegungsfunktionen, sowie die probabilistische Herleitung einer neuen Art von Energiefunktional zur symbolischen Formrekonstruktion. Versuche mit synthetischen, Phantom- und klinischen Datensätzen weisen die Machbarkeit diese Ansatzes nach und zeigen seine Flexibilität.

Stichwörter

Kardiale Computertomographie mit C-Bögen, Dynamische Formrekonstruktion, Level-Set-Methoden, Aktive Konturen

Acknowledgments

I owe a number of "thank yous" to Professor Nassir Navab. Officially referred to as *advisor*, the old-fashioned German term *Doktorvater* is a more accurate description of his way of mentoring Ph. D. students. Encouraging and trusting me from from day one, I was taught more than just scientific work during my five years at CAMP. And although I had different feelings about it at times, I am now very glad that I was able to do quite some teaching assistance, supervision, and project management.

I would also like to thank the other two members of my committee: Professor Rebecca Fahrig and Professor Daniel Cremers were both very helpful and pleasant to coordinate with. They kept the dissertation schedule on time and made a trans-continental, energy-saving defense possible.

Some more people that were critical to this thesis are Dr. Günter Lauritsch and Dr. Jan Boese who supported me and provided valuable input throughout our collaboration, Prof. Dr. Martin Brokate who I got to know as a great lecturer during my undergrad terms and who was able to pinpoint a book from the 1970s – containing a rarely known proof that I needed – amongst the few hundred books in his office, and finally, Jakob Vogel, my colleague with the curiosity of an investigative journalist and the work ethic of a trauma surgeon.

Pursuing a Ph. D. usually means committing oneself to a workplace for several years. Therefore, one better makes sure that the future colleagues are fun to interact with! Luckily, I will always remember the last five years as some of the best ones I ever had. This is due to many people of whom I want to mention a few, omitting doctoral degrees from now on since they (will) all have one: Martin Groher not only challenged and encouraged me during our numerous discussions but also showed me some of the nicest mountain bike routes in his backyard (often proving his superior fitness). Christian Wachinger is an extraordinary office mate – scientifically and personally. Thanks for bringing a smile on my face every day! Darko Zikic was my most loyal car passenger and his sharp-witted mind produced not only great scientific thoughts but also hilarious original jokes and cartoons. Hauke Heibel is one of the few persons you can have a decent argumentative fight as well as a really good time with. Thanks for still having a drink with me, even after heated discussions on software design. Virginie and Pierre Fite-Georgel brought in some style (I yet have to develop) and organized the classiest wedding I attended so far. Moritz Blume was another mountain biking enthusiast whose mind was constantly working (except when he was going uphill on dope music) and Ben Glocker hosted most of the soccer nights that I am missing so much in the U.S. Justina Bauer was not a colleague but a fellow home office worker, always exerting her positive influence - thank you! The list goes on with Selen Atasoy, Maximilian Baust, Marco Feuerstein, Martina Hilla, Martin Horn, Jörg Traub, and many more office, car sharing, sports, and room mates, who made my life so valuable and enjoyable.

The most important people to thank, however, are my parents Erhard and Christine Keil. Their never-ending backing and support gave me the confidence and freedom I am enjoying so much. Thank you for always being there whenever I needed you and for letting go whenever I wanted to.

Contents

I Context of this Work 1

1 Introduction 3
- 1.1 Medical Background . . . 3
- 1.2 Current Cardiac Imaging Technologies . . . 7
- 1.3 Rotational Angiography . . . 11
- 1.4 Coronary C-Arm Reconstruction . . . 12
- 1.5 Contributions of this Work . . . 14
- 1.6 Outline . . . 15

2 Tomographic Reconstruction 17
- 2.1 Computed X-Ray Tomography . . . 17
- 2.2 Analytic Methods . . . 19
- 2.3 Iterative Methods . . . 24
- 2.4 Other Tomographic Reconstruction Methods . . . 27
- 2.5 Existing Approaches in Cardiac Cone Beam CT . . . 28

II Methods 31

3 General Idea 33
- 3.1 Symbolic Reconstruction . . . 33
- 3.2 Motivation for Using Level Sets . . . 34
- 3.3 Related Ideas and Methods . . . 35
- 3.4 Components of a Dynamic Level Set Reconstruction . . . 36

4 Active Contours 39
- 4.1 Lagrangian Modeling . . . 39
- 4.2 Snakes . . . 42
- 4.3 Lagrangian to Eulerian – Classical Level Set Methods . . . 44
- 4.4 Variational Level Set Methods . . . 48
- 4.5 Further Reading . . . 51

5 Shape Reconstruction Using Level Sets 53
- 5.1 Related Work . . . 54
- 5.2 Heuristic Derivation . . . 56
- 5.3 Probabilistic Derivation . . . 57
- 5.4 An Example: Shape from Noisy Silhouette . . . 60

6 Dynamic Level Sets 63
- 6.1 Related Work . . . 63

6.2 Fully Implicit 4D Level Set Function 63
6.3 Implicit Shape and Explicit Motion 64
6.4 An Application: Dynamic Segmentation 67

7 Application to Cardiac Cone Beam Reconstruction 69
7.1 Combining Dynamic Models and Reconstruction Energies 70
7.2 Parametrization, Discretization, and Regularization 71
7.3 Algorithm 76
7.4 Hardware Selection and Runtime Considerations 79

III Evaluation and Discussion 81

8 Experiments 83
8.1 Synthetic Experiments 84
8.2 Phantom Experiments 90
8.3 Real Data 91

9 Conclusion 95
9.1 Discussion of this Work 95
9.2 Future Work 96
9.3 Conclusions for Cardiac C-Arm Reconstruction 99
9.4 Outlook for Medical Image Reconstruction in General 99

IV Appendix 101

A Notation and Nomenclature 103

B Additional Operators 109

C Differentiation and Integration Formulas 111

D Calculus of Variations 115
D.1 Functional Derivative 115
D.2 The Fundamental Lemma in the Calculus of Variations 117
D.3 Deriving Update Directions Using the Calculus of Variations 118

E Calculus for Level Sets 121
E.1 Normal to a Level Set Front 121
E.2 Curvature 122
E.3 Integrating a Function Along a Level Set 123
E.4 Derivative of the Area-minimizing Shape Regularizer 125
E.5 Derivative of the Signed Distance Regularizer 125
E.6 Mollified Versions of Heaviside Function and Dirac Distribution 126

List of (Co-)Authored Publications 131

Bibliography 133

List of Theorems and Definitions

Definition 2.1 Ray transform 18
Definition 2.2 Radon Transform in 2D 20
Theorem 2.3 Inversion of the Radon transform 21
Theorem 2.4 Projection slice theorem in 2D 23

Definition 4.2 Level set function 44

Definition B.1 Adjoint operator to the Radon transform in 2D 109
Definition B.2 Hilbert transform 109
Definition B.3 Fourier transform 109
Definition B.4 Continuous Product 110

Proposition C.1 Derivative of the Norm 111
Proposition C.2 Product Rule 111
Proposition C.3 Integration by parts for functions of scalar arguments 112
Proposition C.4 Integration by parts for scalar fields 112
Corollary C.5 112
Theorem C.6 Divergence theorem 113

Definition D.1 Gâteaux derivative 115
Definition D.2 Functional derivative 115
Corollary D.3 Derivative of integral functionals 116
Theorem D.4 Derivative of the minimum functional I 116
Corollary D.5 Derivative of the minimum functional II 116
Theorem D.6 Fundamental lemma in the calculus of variations 117
Corollary D.7 117

Theorem E.1 Normal to a level set front 121
Theorem E.2 Integration along a level set 123
Corollary E.3 Coarea formula 124

List of Figures

Flip-book movies in the right margin (see Section 8.2 for a description) 1–127

1.1 Heart chambers and blood flow (frontal view) 4
1.2 Schematic illustration of the circulatory system 5
1.3 Various signals of the left ventricle during a cardiac cycle 5
1.4 ECG signal 6
1.5 Imperfect periodicity of cardiac motion 6
1.6 Pathways for accessing the coronary arteries 7
1.7 CT imaging 8
1.8 Angiographic X-ray imaging 9
1.9 PET slice 10
1.10 Transthoracic echocardiography views of a normal heart 11
1.11 IVUS image of a coronary artery 11
1.12 Acquisition setting for cardiac rotational angiography 13

2.1 Illustration of the ray transform 18
2.2 Illustration of the Radon transform 20
2.3 Radon transform of a synthetic image 21
2.4 Effect of filtering before backprojection. 22
2.5 Illustration of Kaczmarz' method in 2D with $\lambda(k) = 1$ 26

3.1 Schematic illustration of the inconsistency vs. baseline trade-off 34
3.2 Components of a dynamic symbolic level set reconstruction from angiographic projections 37

4.1 Explicitly modeled contour 40
4.2 Representation of a curve by a discretized level set function 45
4.3 An implicitly evolved contour in 2D 45

5.1 Illustration of Problem 5.1 54
5.2 False positive and false negative reconstructions. Note that the image $\tilde{I}$ is assumed to contain high intensity or probability values for projections of vessels. Therefore a black pixel (representing low values) is a mismatch with the reconstructed (false positive) vasculature $\{\Phi < 0\}$ (represented by black voxels). Analogously, high image values and rays with positive Φ values characterize false negatives. 56
5.3 Two-dimensional shape reconstructions 61
5.4 Three-dimensional reconstructions 62

6.1 Dynamic level set as series of volumes 64
6.2 Dynamic level set as warped reference volume 65
6.3 Motion parametrization/discretization 66
6.4 Dynamic segmentation of a 4D CTA data set 68

7.1 Level set resolution 72
7.2 Interpolation of level set updates 78

8.1 Imaging and reconstruction process for synthetic data with global, rigid motion . 86
8.2 Exemplary overlays for evaluation of shape errors for the synthetic data sets with deformable, periodic motion 88
8.3 Illustration of Dice's similarity coefficient / overlap ratio 89
8.4 Error evaluation of one reconstructed volume for the phantom data set 91
8.5 4D error measure for the phantom data set 92
8.6 Results for first tests on real angiographic RA data 94

A.1 Planes in human anatomy 107

D.1 Illustration of the derivative of the minimum functional 117

E.1 Osculating circle 122
E.2 Planes and curves of principal curvature of a surface point 122
E.3 Mollified versions of the Heaviside step function and the Dirac distribution . . . 128

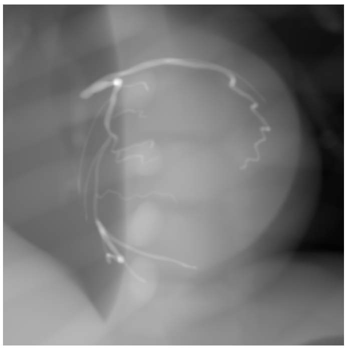
List of Tables

1.1 Average coronary artery lumen diameters of the normal population 13

5.1 Quantitative evaluation of 2D reconstruction experiments 62

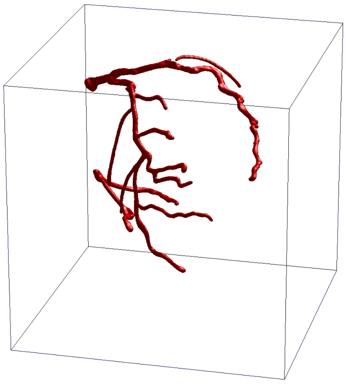
8.1 Problem dimensions for the synthetic experiment with rigid, non-periodic motions 85
8.2 Comparison of motion reconstruction errors 86
8.3 Problem dimensions for the synthetic experiment with deformable, periodic motions 87
8.4 Comparison of reconstruction errors for two types of synthetic data sets at six different noise levels. 89
8.5 Problem dimensions for the phantom experiment 90
8.6 Problem dimensions for the real experiment 93

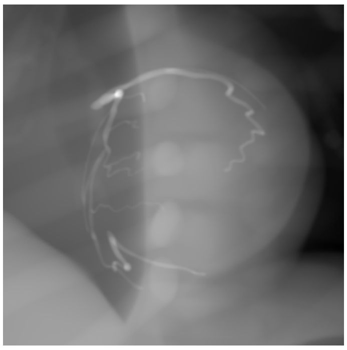
A.1 Typefaces and Notation 103
A.2 Variables, Sets, and Functions 104
A.3 Acronyms 105
A.4 Names and Abbreviations of the Coronary Arteries' Branches 107
A.5 Terminology 107

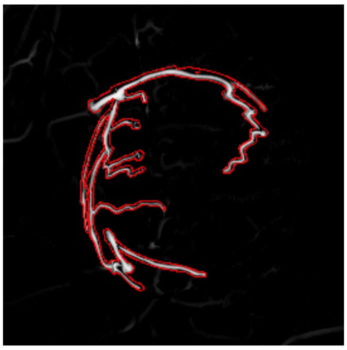


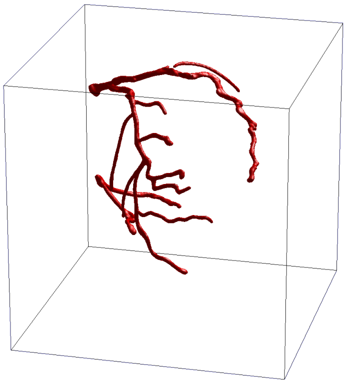
Part I

Context of this Work









Chapter 1

Introduction

This thesis strives to present novel methods for the reconstruction of dynamic shapes from a series of projection images acquired by a single moving camera. The methods that are developed for this setting are applicable in a very general setting and are based on work in the image segmentation and the computer vision domain. The original motivation for this problem, however, was the 3D reconstruction of the coronary arteries from angiographic X-ray sequences as they are routinely acquired in catheter laboratories (informally called *cath labs*). These interventions would benefit tremendously from the incorporation of 3D information compared to the state-of-the-art 2D images.

A straightforward reconstruction from C-arm data, however, is hampered by the "motion blur" resulting from the relatively long acquisition time of several seconds. But this amount of time is required for taking enough images for a 3D reconstruction using a C-arm – the standard imaging device for cardiac interventions. Considering that a normal human heart beats at least once per second, one must not ignore the cardiac motion but is instead forced to estimate and compensate for it during the reconstruction.

But before going into details of the methods developed throughout this thesis, we will first focus on this specific application's domain and review the medical background as well as the image reconstruction approaches generally employed there. After developing a new framework for 4D shape reconstruction in the central part of this thesis, we will return to the initial application for testing and evaluation purposes in Part III.

1.1 Medical Background

This section should give a very limited but focused overview of the cardiac anatomy and its relevant physiology, the diseases associated with its blood supply, and the minimally invasive procedures used to cure them.

1.1.1 Cardiac[1] Anatomy and Physiology

The heart is essentially a combination of two pumps, each consisting of two chambers (atrium and ventricle) that are compressed by the contraction of the surrounding muscles in the myocardium. See Figure 1.1 for an illustration of the heart chambers and their internal blood flow. The pressure differences created by the heart keep the blood circulating through the body (where it is

---

[1] The term *cardiac* originated from the Greek καρδιά for *heart* and has the meaning *heart-related*.

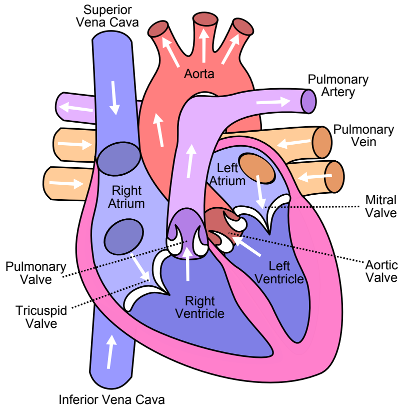


Figure 1.1: Heart chambers and blood flow (frontal view). (Image source: `http://commons.wikimedia.org/wiki/File:Diagram_of_the_human_heart_(cropped).svg`. Original author: Eric Pierce. License: `http://creativecommons.org/licenses/by-sa/3.0/`.)

deoxygenated) and through the lungs (for oxygenation). This overall cycle is illustrated in Figure 1.2. The contractions are initiated by electrical signals first triggering atrial and then ventricular contraction. This, in combination with the four cardiac valves, results in the required pressure gradients (see Figure 1.3). The electric signals can be recorded on an electrocardiogram (ECG) which is not only used for diagnosis itself but can also give valuable information about the current cardiac phase during image acquisition. In the latter case, the standard approach is to use the *R peaks* only (see Figure 1.4), normalizing each cardiac cycle and mapping every point in time to a periodic RR-interval with percentage values. Such information about the cardiac phase is of crucial importance during 3D reconstruction from data with a low temporal resolution. It represents prior knowledge that can be used for coping with the component of the cardiac motion that results from the contractions. Assuming a totally periodic cardiac motion, however, is not possible. Even in the absence of breathing motion, the coronary arteries do not reposition exactly between adjacent heart beats according to Achenbach [1] (see also Figure 1.5). The additional breathing and patient motion are usually minimized by asking the patient to hold breath and keep still.

1.1.2 Coronary Heart Disease

Among the cardiovascular diseases (CVD), the coronary heart disease (CHD) is the primary cause of death killing 7 million people per year worldwide, closely followed by stroke, see the WHO's heart and stroke atlas [169]. CHD is actually the primary cause of death since the early 1990's and now accounts for approximately one third of all deaths in the U.S. (see the AHA statistics [103]). It is generally more prominent in countries with a high life expectancy but most severely affects developing countries in terms of *healthy years of life lost* (see also [169]).

CHD is often caused by atherosclerosis leading to coronary stenosis which describes obstructive lesions narrowing the lumen of the coronary arteries. In

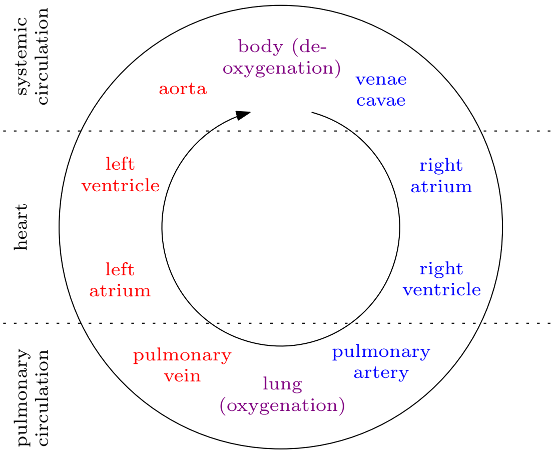

systemic circulation
heart
pulmonary circulation
body (de-oxygenation)
aorta
venae cavae
left ventricle
right atrium
left atrium
right ventricle
pulmonary vein
pulmonary artery
lung (oxygenation)

Figure 1.2: Schematic illustration of the circulatory system

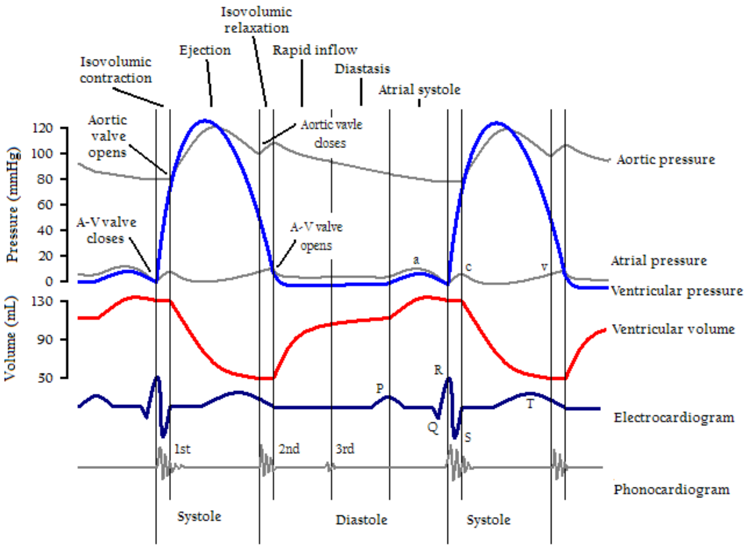
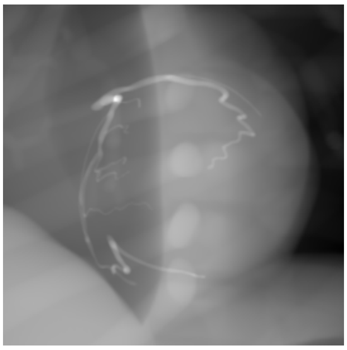
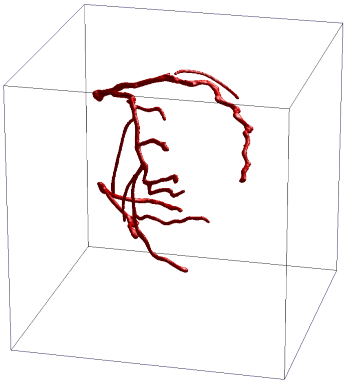


Figure 1.3: Various signals of the left ventricle during a cardiac cycle. (Image source: `http://commons.wikimedia.org/wiki/File:Cardiac_Cycle_Left_Ventricle.PNG`. Author: User DestinyQx. License: `http://creativecommons.org/licenses/by-sa/2.5/`)





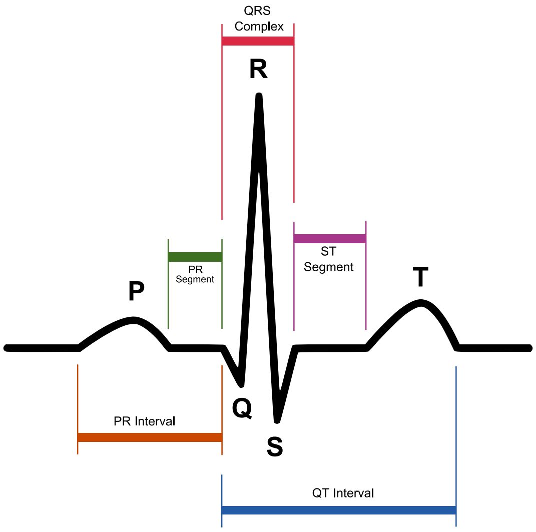

QRS
Complex
R
PR
Segment
ST
Segment
P
T
Q
S
PR Interval
QT Interval

Figure 1.4: ECG signal. (Image source: `http://commons.wikimedia.org/wiki/File:SinusRhythmLabels.svg`. License: Unrestricted.)

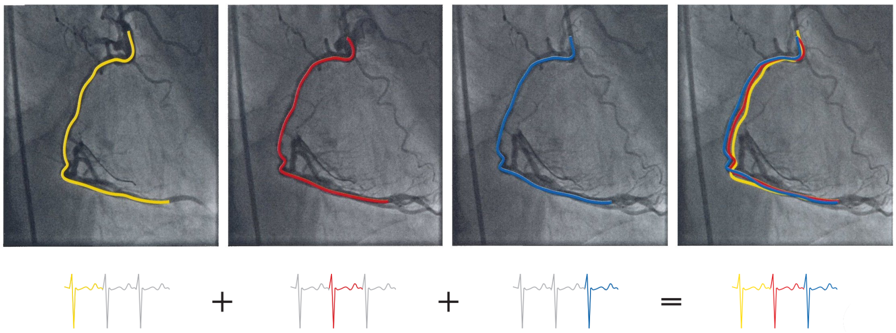


Figure 1.5: Imperfect periodicity of cardiac motion, visualized by overlaying angiograms from several heart beats. (Image derived from original work used by courtesy of Achenbach [1].)

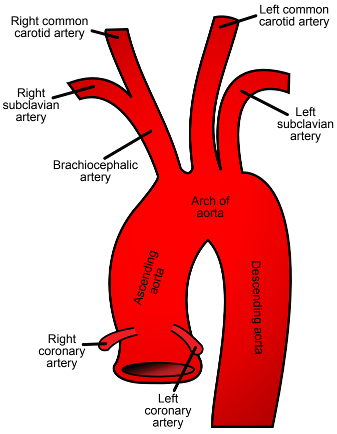
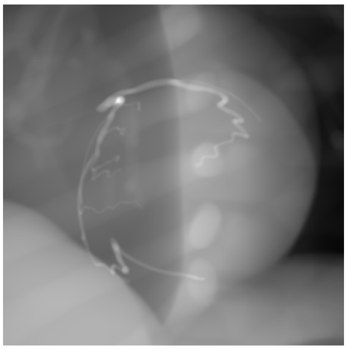
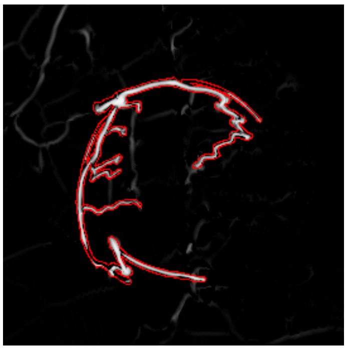
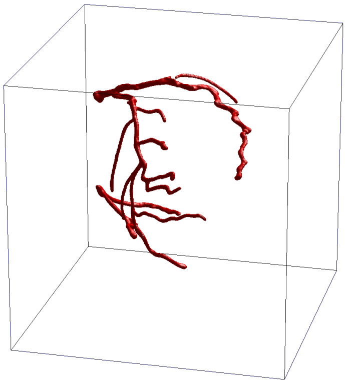

Right common carotid artery
Left common carotid artery
Right subclavian artery
Left subclavian artery
Brachiocephalic artery
Arch of aorta
Ascending aorta
Descending aorta
Right coronary artery
Left coronary artery

Figure 1.6: Pathways for accessing the coronary arteries. (Image source: `http://commons.wikimedia.org/wiki/File:Gray506.svg`. Author: User Rhcastilhos. License: Public domain.)







their acute form, such stenosis cause the acute coronary syndrome (ACS) manifesting in chest pain and other symptoms. The narrowing (or even total occlusion) of coronary arteries leads to a decreased blood supply of the myocardium (the cardiac muscles), impeding the regular function of the heart.

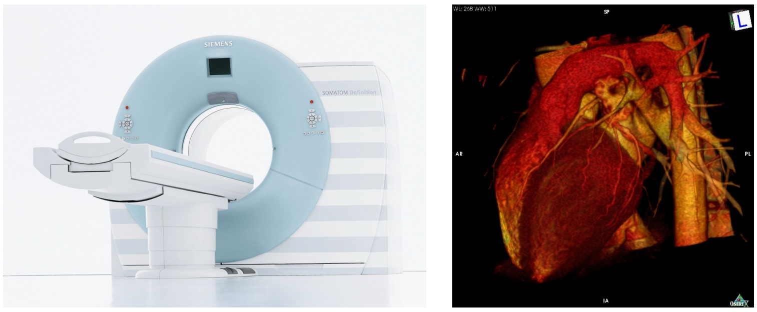
1.1.3 Coronary Interventions

In contrast to open cardiac surgical procedures (like bypass surgery), this thesis focuses on catheterized interventions performed in cath labs. These interventions are mainly performed to widen narrowed coronaries using an inflatable balloon (angioplasty) and to place stents. In order to perform such percutaneous coronary interventions (PCI), a tube (called *catheter*) is inserted through the femoral artery (in the groin) or the radial artery (in the wrist) and navigated to the point in the ascending aorta where the coronary arteries branch off (see Figure 1.6). Other catheterization procedures include ablations and the widening or replacement of valves. In some of these cases, the catheter is inserted through the venous system.

A prerequisite for PCI is the availability of real-time imaging for navigating through the vascular system to the heart and within the heart's own vasculature. Some of the cardiac imaging modalities will be discussed in the next section, esp. with respect to their applicability in a cath lab setting.

1.2 Current Cardiac Imaging Technologies

Depending on the medical history of a patient, different imaging modalities are used for diagnosing cardiac problems. All of the following methods have their pros and cons and all-purpose device does not exist.



(a) CT scanner (b) Volume rendering of CTA data

Figure 1.7: CT imaging. (Image (a) courtesy of Siemens AG. Image (b) reproduced by courtesy of Antoine Rosset, M.D., LaTour Hospital, Geneva, Switzerland)

1.2.1 Cardiac CTA

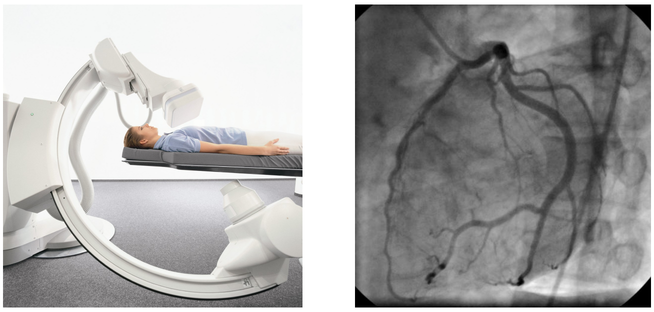
Computed tomography (CT, from the Greek τόμος meaning *slice*), when employed for imaging vessels, is denoted by the acronym CTA which stands for *CT angiography* (from the Greek ανγειον for *vessel*). Since the radio density of blood is similar to that of surrounding tissue, contrast dye (usually containing iodine atoms for X-ray angiography) has to be injected into the vessels of interest. In the case of CTA, this is usually done via a small peripheral vein that is easily accessible. This method of contrasting vasculature is therefore less invasive compared to the catheter-based injection in angiography. The raw data acquired along a helical source trajectory (see Figure 1.7(a) for a CT scanner) is then reconstructed to obtain a 3D volume of radio density distributions. These attenuation coefficients are given on the Hounsfield unit (HU) scale which is defined by

$$\begin{aligned} \mathrm{HU}(\mathrm{air}) &= -1000 \\ \mathrm{HU}(\mathrm{water}) &= 0 \quad . \end{aligned}$$

The 3D volumes have to be somehow projected to 2D for visualizing them on a screen. The usual ways to achieve this dimensionality reduction are

- visualization of transversal slices (corresponding to the orientation used in the reconstruction algorithm),
- visualization of other slices (sagittal, coronal, or oblique), called multiplanar reformatting (MPR),
- volume rendering (see Figure 1.7(b)), and
- maximum intensity projection (MIP).

For cardiovascular diagnosis, CTA is primarily used for ruling out intravascular lesions when a patient's medical history does not indicate such diseases. (Otherwise, the probable subsequent intravascular intervention also enables an integrated diagnosis and treatment, making a prior CT superfluous.) Due to the improvements of current CT systems in terms of speed and X-ray dose,



(a) C-arm

(b) Contrasted coronaries in angiogram

Figure 1.8: Angiographic X-ray imaging. (Image (a): Courtesy of Siemens AG. Image (b): Source: `http://commons.wikimedia.org/wiki/File:Hk_coronary_bionerd.jpg`. Author: Cath lab at Charité Mitte, Berlin, Germany. License: `http://creativecommons.org/licenses/by/3.0/`)

there is a trend to routinely perform CTA, esp. in the United States. The main advantages of CTA are the accuracy, acquisition speed, low level of invasiveness (compared to catheter-based methods), and data richness. The disadvantages are the comparably high dose of ionizing radiation and the additional time and work flow steps needed for this purely diagnostic tool. The intra-interventional use is tempered by the size of the apparatus (making it an *exclusive* imaging device) as well as the unacceptable high dose if it were to be used for navigation.

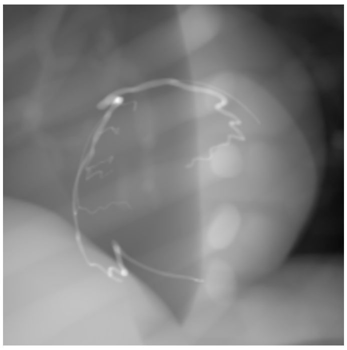
1.2.2 Angiography / Fluoroscopy

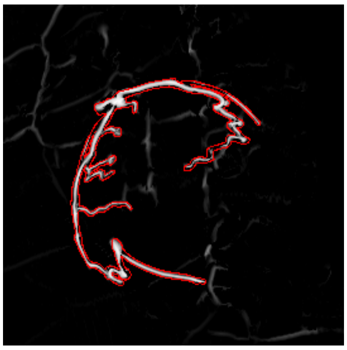
For obtaining contrasted X-ray projections of the coronary vasculature, one has to access these vessels through the body's arterial or venous system with a catheter in the same way as for an intervention. Injecting contrast material into the vasculature of interest during a fluoroscopic (movie-like) acquisition of X-ray projections (see Figure 1.8(b) for one frame) allows the cardiologist to visualize the vessel lumen and also qualitatively estimate the blood flow. Due to its real-time nature, this modality can be used for navigation but also for diagnosis. The latter, however, is hampered by the projective nature of the image data making it difficult to estimate lengths, diameters, and angles from the images. The X-ray devices used for such acquisitions are usually mounted on a steerable, C-shaped frame called *C-arm* (see Figure 1.8(a)). This enables the cardiologist to easily acquire projections from several directions in order to better cope with the foreshortening and overlapping of vessels.

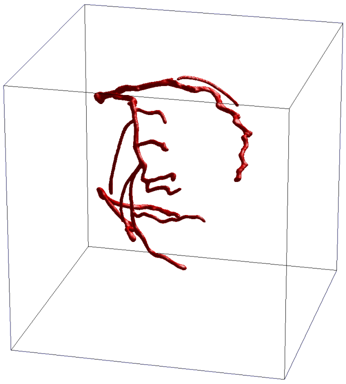
1.2.3 Other Modalities

MRI

Magnetic resonance imaging (MRI) or *nuclear magnetic resonance* (NMR) imaging is a non-invasive technique using the electromagnetic properties of atomic nuclei. Although having a very low temporal resolution, the modality's non-invasiveness enables to acquire a lot of data without harming the patient.







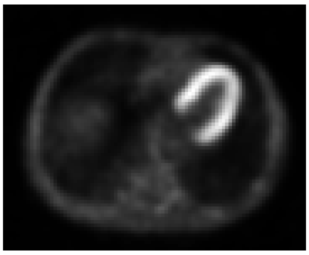


Figure 1.9: PET slice. (Image reproduced by courtesy of Antoine Rosset, M.D., LaTour Hospital, Geneva, Switzerland)

In cardiology, cine-MR sequences created this way provide a dynamic display and evaluation of the cardiac motion and perfusion. The complicated setup and high costs, however, make it a specialized modality with limited applications.

Emission tomography (PET/SPECT)

The nuclear medicine imaging techniques *positron emission tomography* (PET) and *single photon emission computed tomography* (SPECT) are both using radiation emitted from radioactive tracer injected into the patient's body. The anatomy or physiology imaged can be influenced by the type of tracer that is injected. The distinguishing properties of PET and SPECT are mainly the type of radiation emitted and the acquisition geometry. PET uses fixed rings detecting the gamma photons emitted in opposite directions from the annihilation location of positrons originally emitted by the tracer. SPECT is based on rotating cameras measuring the directly emitted gamma radiation in a projective manner.

In cardiology, SPECT is used for myocardial imaging, e. g., for testing for an ischemic heart disease. But the coronary vasculature cannot be analyzed directly with PET or SPECT due to both modalities' relatively low resolution (see Figure 1.9). The myocardial imaging only provides indirect evidence for reduced blood supply.

Ultrasound

Transthoracic echocardiography (TTE) and transesophageal echocardiography (TEE) are both imaging modalities using ultrasound reflections (echos). The former method is non-invasive since the probe is placed on the patient's chest while the latter is considered invasive because it requires to enter the ultrasound probe through the patient's esophagus. TTE and TEE enable a faster and less invasive diagnosis of the heart's condition but cannot be used for an intervention inside the coronaries. See Figure 1.10 for some examples of cardiac ultrasound images. There are also transducer arrays available that enable a 3D image acquisition, particularly useful for understanding valvular and congenital malformations. Doppler echocardiography is performed for assessing valve function, cardiac output, and ventricular filling ratios. Note that the associated velocity measurements are usually based on the *phase shift*, not on the Doppler effect (which is synonymous for the *frequency shift*).

Intravascular ultrasound (IVUS) is different in that the ultrasound probe is inserted via a catheter *inside* the vasculature, e. g., the coronary arteries, making it as invasive and time-consuming as other catheterized procedures. It

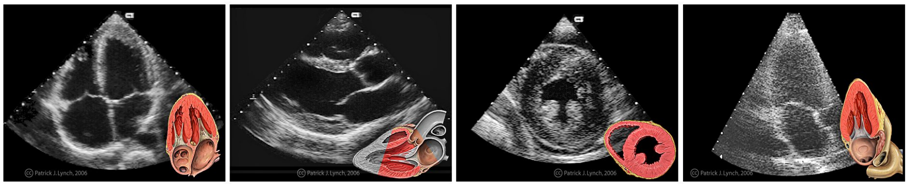


Figure 1.10: Transthoracic echocardiography views of a normal heart. (Derived work. Image source: `http://commons.wikimedia.org/wiki/File:Heart_normal_tte_views.jpg`. Authors: Patrick J. Lynch, medical illustrator and C. Carl Jaffe, MD, cardiologist. License: `http://creativecommons.org/licenses/by/2.5/`.)

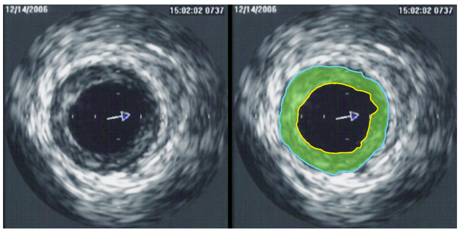
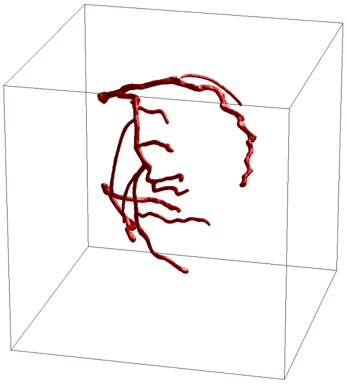


Figure 1.11: IVUS image of a coronary artery, with color coding on the right (atherosclerotic plaque burden shaded in green). (Image source: `http://commons.wikimedia.org/wiki/File:IVUS_of_CAD_(1).png`. Author: User Ksheka. License: `http://creativecommons.org/licenses/by-sa/2.5/`.)

can be used to accurately assess plaque volume in the vessel wall as well as the degree of stenosis (see Figure 1.11).

1.3 Rotational Angiography

Rotational angiography (RA) is a more recent imaging technique based on a mounted and motorized C-arm acquiring a series of X-ray projections while rotating around the patient. A prior calibration of the acquisition geometry (see, e. g., Fahrig et al. [43], Fahrig and Holdsworth [42] for image-based calibration methods and accuracy and Navab et al. [118] for the calibration and reconstruction using projection matrices) then enables the 3D reconstruction of the imaged anatomy. The quality that can be achieved with C-arms is now similar to that of conventional CT scanners – mainly thanks to the development of high-resolution flat panel detectors. The big advantage compared to conventional systems is the open shape of the device which allows to use it intra-operatively. However, due to the relatively slow motion of the C-arm, only static objects can be reconstructed without motion blur. This currently limits RA reconstructions to still anatomy like the brain, the abdomen, or the extremities. Even for those applications, the patient has to remain still (and hold her breath in case of abdominal imaging) during the acquisition which typically takes 5 s to 10 s.



1.4 Coronary C-Arm Reconstruction

Benefits

As mentioned before, C-arms are already the modality of choice for coronary interventions. They are used in a fluoroscopic mode in cath labs for diagnosis and treatment. In this setting, a near-time 3D reconstruction would provide valuable information to the cardiologist in addition to the real-time 2D navigation data. Such volumetric information would allow an exact assessment and measurement of narrowings without the limitations of projective imaging such as foreshortening and overlap. Even the real-time 2D imaging would benefit from an intra-interventional volumetric reconstruction which helps to find an optimal viewing direction (Green et al. [59]). This would not only improve diagnosis but also treatment (e. g. in terms of stent choice). Moreover, post-treatment evaluation and quantitative coronary angiography would also benefit from more exact measurements. In addition to improving the accuracy and making coronary interventions less operator-dependent, RA was shown to decrease the amount of contrast agent needed, the radiation exposure, and the acquisition time. See Maddux et al. [106, 107] and Garcia et al. [51] for such considerations.

Problems

But compared to other reconstruction methods, a coronary reconstruction algorithm has to address some additional issues, making it a particularly difficult problem:

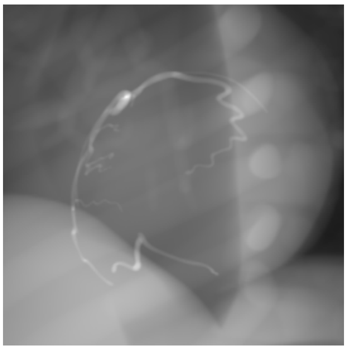
The breathing motion is usually minimized by asking the patient to hold breath for several seconds. This, however, introduces a new problem in the case of coronary reconstruction: Very often, the heart rate first slows down and then accelerates during a continued breath hold. This is problematic since an instable heart rate further complicates a reconstruction by decreasing the temporal resolution of gating-based algorithms (Lauritsch et al. [99]). The expiration state was found to be better suited than deep inspiration for minimizing this effect in most cases (see, e. g., Lauritsch et al. [98], Movassaghi et al. [113]).

The cardiac motion, on the other hand, cannot be avoided at all. It also cannot be ignored since it takes place on a much faster time scale than the acquisition. Instead, various strategies (based on ECG data and/or image matching) exist for estimating the motion state of the heart and incorporating this knowledge into the reconstruction. See Figure 1.12 for an illustration of the acquisition setting.

Such motion modeling has to take into account another issue that arises out of the irregularity of the cardiac motion in combination with the reconstruction detail required to assess the coronary vasculature: The re-positioning “error” is on the order of the diameters of the coronaries (see Table 1.1). This implies that the cardiac motion cannot be assumed to be perfectly periodic.

Existing approaches and new approach

Traditional tomographic reconstruction methods like filtered backprojection or algebraic reconstruction (see Chapter 2) have to be modified to be applicable for cardiac cone beam CT. Other attempts for the reconstruction of the coronaries are based on the assumption that the shape information is easier to obtain than a fully tomographic reconstruction of the attenuation information. This



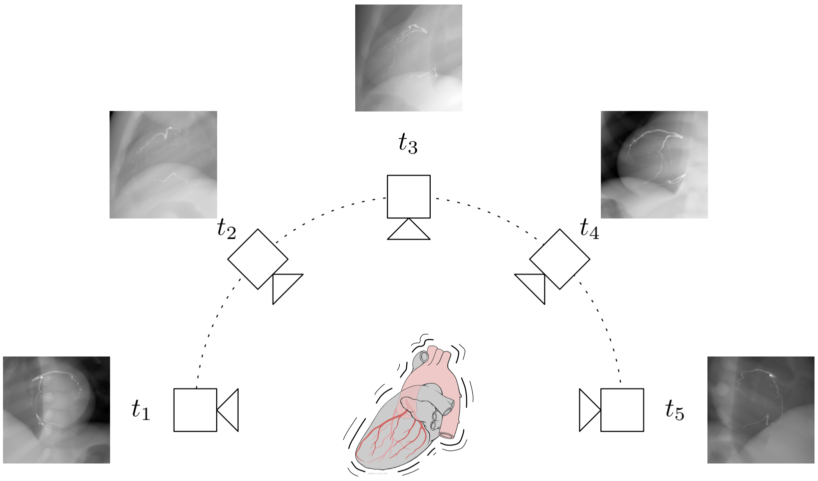
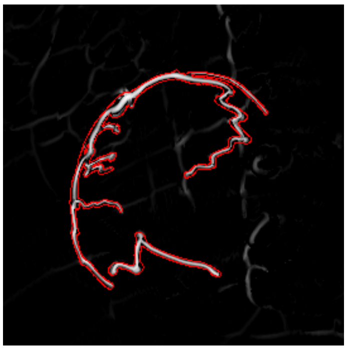


Figure 1.12: Acquisition setting for cardiac rotational angiography (Image based on original drawing available at `http://commons.wikimedia.org/wiki/Image:Heart_left_lateral_coronaries_diagram.svg`. Original authors: Patrick J. Lynch, medical illustrator; C. Carl Jaffe, MD, cardiologist. License: `http://creativecommons.org/licenses/by/2.5/`)



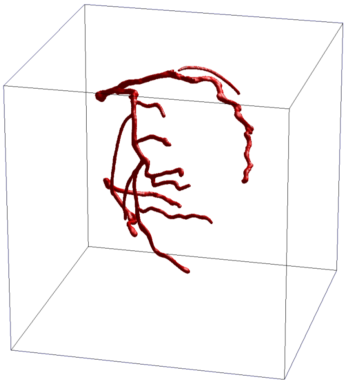


Table 1.1: Average coronary artery lumen diameters of the normal population. Generally (including men with left ventricular hypertophy and men with dilated cardiomyopathy, all right-dominant) one can expect a range of diameters from (0.7 ± 0.2) mm to (4.9 ± 0.7) mm. (All data extracted from Dodge et al. [38]. Standard deviations range from 0.4 mm to 0.6 mm for all data given below.)

Arterial branch[a]		Coronary dominance[b] right/balanced	 left
RCA	proximal	3.9 mm	2.8 mm
	distal	3.1 mm	1.1 mm
LM		4.5 mm	
LAD	proximal	3.7 mm	
	distal	1.9 mm	
LCx	proximal	3.4 mm	4.2 mm
	distal	1.6 mm	3.2 mm

---

[a] See Table A.4 for abbreviations of the branch names.

[b] Coronary dominance is defined by the artery supplying the posterior interventricular artery (PIV), a.k.a. the posterior descending artery (PDA). Approximately 80 % to 90 % of the population have a right or balanced dominance.

lead to the development of dynamic triangulation methods using preprocessed (vessel-enhanced) images. All these approaches are summarized in more detail in Section 2.5. None of them is in clinical use routinely, probably because this tough reconstruction problem still is not solved robustly enough.

This is why we developed some novel ideas on how to model the problem. The methods to be presented in Part II share many of the ideas of other existing approaches but have been designed with some central objectives in mind:

- The resulting algorithm should be as flexible as possible enabling the incorporation of as much information and prior knowledge as is necessary for obtaining a well-posed reconstruction problem.
- The robustness of a volumetric reconstruction should be combined with the dimensionality reduction[2] achieved by reconstructing symbolic or shape information only.
- The reconstruction of shape and motion is not separable. Neither of the two can be correctly estimated without knowing the other. Such coupled problems can usually be solved better if treated simultaneously instead of consecutively.
- Assumptions and prior knowledge should be included as soft constraints, using as much additional input as possible without over-restricting the solution to ideal cases.

We have chosen to use level sets as a shape model for representing the coronary arteries. Although it is challenging to develop suitable reconstruction energies, there are various reasons for favoring them over explicit shape models like point sets or snakes. The other modeling problem addressed in this thesis is the "dynamization" of these level sets – traditionally only used to represent static objects. In this case, we opted for an explicit representation of the motion which greatly simplifies the incorporation of prior knowledge.

The various arguments, that lead to the choices summarized briefly before, are all discussed in detail in Chapter 3, after a review of the existing approaches for cardiac cone beam reconstruction in Chapter 2.

1.5 Contributions of this Work

The main contributions of this thesis are the novel methods developed for dynamic shape reconstruction from probabilistic projections of arbitrary shape, motion, and projection geometry. This includes two novel mathematical formulations, one for obtaining dynamic shape models and the other for reconstructing such shapes from a series of probabilistic projection images.

The new formulation for dynamic shape combines a static, implicit level set shape model with an explicit motion model. The implicit shape definition ensures the flexibility of the model – a feature which is of great value given the unknown and highly varying configuration of coronary vasculature. The explicit motion modeling, on the other hand, allows the incorporation of some prior knowledge, esp. about the time-continuity of the reconstruction.

---

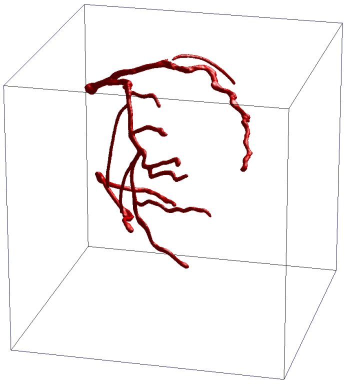
[2] The amount of input data required increases exponentially as the physical dimension of the reconstruction problem increases ("curse of dimensionality", see e. g., Bishop [9]).

The other crucial part is the definition of an appropriate energy functional, chosen so that its minimization yields the desired reconstruction. Contrary to existing, volume-based formulations, we developed a new, ray-based similarity measure. Although the resulting functional is mathematically and algorithmically challenging to optimize, it allows a more robust reconstruction of *dynamic* scenes by not assuming any level of consistency in the input data.[3]

These methods – that are related to prior work in computer vision – have been developed with a medical application in mind. Although the results for the dynamic reconstruction of coronary artery trees from rotational angiographic data are quite promising, the intention and value of this work is the introduction of new methods for dynamic shape reconstruction. The advantageous properties of the new formulations, namely the versatility in shape and motion modeling as well as the robustness, make them a promising ingredient for a variety of reconstruction problems.

1.6 Outline

- Part I provides the context for this thesis. After the introduction in **this chapter**, the related work in tomographic reconstruction is reviewed in **Chapter 2**.
- Part II mostly contains the original work of this thesis. After formalizing the setting and describing the ideas in **Chapter 3**, we give a short introduction to the theory of active contours in **Chapter 4**. Based on active contours, we present novel shape reconstruction functionals as well as dynamic shape models in **Chapters 5** and **6**, resp. These new models are then combined to obtain formulations for a dynamic shape reconstruction in **Chapter 7**.
- Part III shows the results obtained in the experiments (in **Chapter 8**) and draws conclusions on the methods developed in this work as well as in a broader sense on expected developments in medical image reconstruction in general (in **Chapter 9**).
- Part IV is the appendix. Apart from the notation and nomenclature in **Appendix A** it also contains all the mathematical definitions and theorems used in this thesis in the remaining **Appendices B** to **E**.

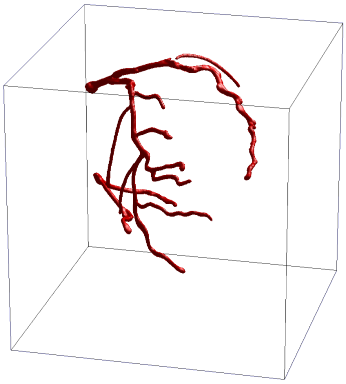


---

[3] Compare this to volumetric-driven reconstruction functionals which have to have some threshold for the number of projections that have to agree on seeing an object. This threshold would become higher with the higher consistency achieved by the simultaneous motion optimization.

Chapter 2

Tomographic Reconstruction



This chapter summarizes the methods that are “traditionally” employed for tomographic reconstruction as well as those that have been developed for the specific application of cardiac cone beam CT. The formulations and models developed during this thesis, however, are mostly based on methods not commonly used in medical image reconstruction.

Remark. A further area of related work is the enhancement of vessels in angiographic images (needed as a preprocessing step in shape reconstruction approaches). The discussion of this well-established area of research is out of the scope of this thesis and the reader is referred to the literature (e. g. see [89]). Furthermore, an introduction to the theory of active contours is out of the scope of the collection of methods in this chapter and postponed to Chapter 4, right before the related methods are used in subsequent chapters.

2.1 Computed X-Ray Tomography

Most of the methods discussed in the following sections are used for tomographic reconstruction where one seeks to recover the attenuation field μ (measured in units of inverse length like cm^{-1}) from the radiation intensities

$$\boxed{I(L) = I_0 \cdot \exp\left(-\int_L \mu(\boldsymbol{X})\ \mathrm{d}\boldsymbol{X}\right)} \tag{2.1}$$

measured on the X-ray detector for projection lines L. Equation (2.1) expresses that the radiation I is subject to an *exponential decay* with the attenuation $\mu(\boldsymbol{X})$ as variable *decay constant*. Note that this is only a model of the real process of X-ray attenuation. Even major effects like beam hardening (see e. g. Stonestrom et al. [160] for an attenuation equation modeling multi-spectral beams) and scattering (esp. prominent in cone beam CT) are not included in this approximation. The *transmission* value – the ratio of detector intensity I to the (also known) source intensity I_0 – is then

$$\frac{I(L)}{I_0} = \exp\left(-\int_L \mu(\boldsymbol{X})\ \mathrm{d}\boldsymbol{X}\right) \quad ,$$

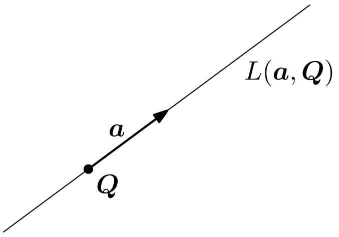


Figure 2.1: Illustration of the ray transform (Definition 2.1)

and applying a negative logarithm, one finally obtains

$$\boxed{-\ln\frac{I(L)}{I_0} = \int\limits_L \mu(\boldsymbol{X})\ \mathrm{d}\boldsymbol{X}} \tag{2.2}$$

for the dimensionless attenuation integral over the line L. This attenuation integral can also be expressed using the *ray transform*:

Definition 2.1 (Ray transform)
Let $\mu\colon \mathbf{R}^n \to \mathbf{R}$ *be a function with suitable smoothness properties and let*

$$L(\boldsymbol{a},\boldsymbol{Q}) = \{\boldsymbol{X} \in \mathbf{R}^n\colon \boldsymbol{X} = \boldsymbol{Q} + s\boldsymbol{a}, s \in \mathbf{R}\}$$

be a line in $\mathbf{R}^n$*, parametrized by a point* $\boldsymbol{Q}$ *on the line and the normalized line direction* $\boldsymbol{a}$ *(see Figure 2.1). The* ray transform *(also called* X-ray transform*)* $\mathcal{P}$ *of* μ *along the line* L *is then given by*

$$\begin{aligned}(\mathcal{P}\mu)(L) &= \int\limits_L \mu(\boldsymbol{X})\ \mathrm{d}\boldsymbol{X}\\ = (\mathcal{P}\mu)(\boldsymbol{a},\boldsymbol{Q}) &= \int\limits_{-\infty}^{\infty} \mu(\boldsymbol{Q}+s\boldsymbol{a})\ \mathrm{d}s \quad .\end{aligned}$$

This operator is closely related to the *Radon transform* which will be introduced later. In 2D they are identical.

Remark. Note that in our application (cone beam reconstruction) the projection geometry is modeled by a series of projection operators $\boldsymbol{P}(\cdot,t_l)\colon \mathbf{R}^3 \to \mathbf{R}^2$ so that the lines of integration are given by $L(\boldsymbol{x},t_l) = \boldsymbol{P}^{-1}(\boldsymbol{x},t_l)$ (for the line corresponding to pixel $\boldsymbol{x}$ in projection no. l).

Using Definition 2.1 and the shorthand notation

$$g(L) := -\ln\frac{I(L)}{I_0}$$

for the normalized measurements we are now able to rewrite (2.2) using the ray transform defined above to obtain

$$\boxed{(\mathcal{P}\mu)(L) = g(L) \quad \forall\, L\ .} \tag{2.4}$$

Assuming (2.2) perfectly models the physical process of X-ray attenuation and intensity measurement[1], this defines a relation between the unknown attenuation μ and the measurements g. The methods traditionally employed for the tomographic reconstruction problem of finding μ from (2.4) can be grouped into

- analytic (filtered back projection and Fourier reconstruction),
- iterative (ART, SART, . . .), and
- probabilistic (expectation maximization)

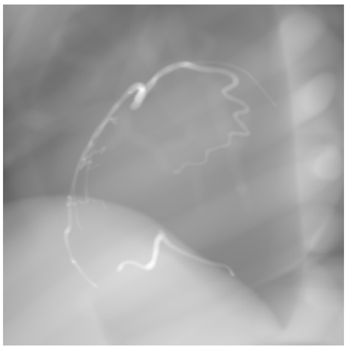
methods, where the latter two share the same order of precedent discretization and subsequent inversion. Analytic methods, in contrast, are based on an analytic inversion of the operator $\mathcal{P}$ in (2.4) and discretization is performed in the very last step.

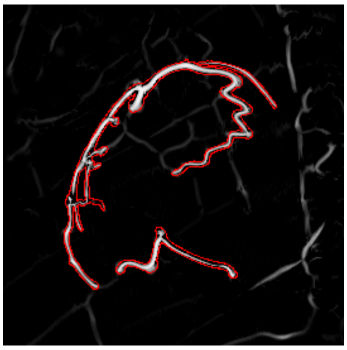
As mentioned before, this chapter can only give a brief introduction to reconstructions methods known from the literature. The interested reader is pointed to textbooks like Natterer [115], Natterer and Wübbeling [116], Herman [65], Kak and Slaney [83] and Buzug [19]. For a historical perspective, one might also look at the original paper by Hounsfield [69] on the first system assembled by Sir Godfrey Newbold Hounsfield[2] or the independent work on analytic reconstruction by Cormack [29, 30]. These two pioneers were jointly awarded the Nobel Prize for Physiology or Medicine in 1979 for their independent work and achievements in computed tomography. Apart from them many others have independently and with different applications in mind worked on tomographic reconstruction. Most notably, Radon laid a theoretical foundation for analytic reconstruction in [132] which was, like other prior work in analytic reconstruction, unknown to Cormack until years after he published his work. Also Alessandro Vallebona should be mentioned for inventing the first *mechanical* tomographic system already in 1930. By moving X-ray source and film in opposite directions, projections were superimposed with only a single plane (through the center of rotation) accumulating on the film. More historical remarks can be found in Natterer and Wübbeling [116, sec. 3.9].

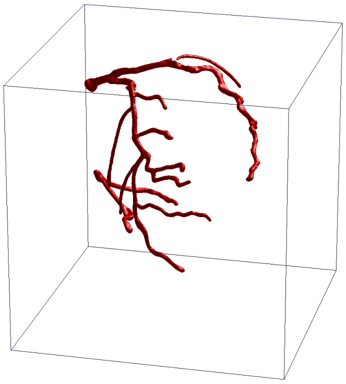
Remark. As an anecdotal note, it should be mentioned that the breakthrough of CT was probably accelerated by *The Beatles*, who, through their unanticipated success, flushed their record label EMI (Electric & Musical Industries Ltd.) with funds. This enabled EMI's Central Research Laboratories to fund Sir Hounsfield's initial prototype. With additional support from Britain's Department of Health and Social Security (DHSS) after the decline of the music industry, Hounsfield was then able to bring the first so-called *EMI scanner* to market by 1972, only five years after he started the project.

2.2 Analytic Methods

We will introduce the so-called *analytic methods* beginning with Johann Radon's 2D-formulation and inversion formula deduced in 1917. Other methods like filtered back projection (FBP), its generalization to cone-beam tomography







---

[1] Note that we neglected many effects like scattering, beam hardening, and noisy intensity measurements here.

[2] Hounsfield actually employed an iterative reconstruction although the analytic methods picked up quite soon afterwards and become an industry standard due to their lower computational cost (only to be challanged by itearive methods nowadays again).

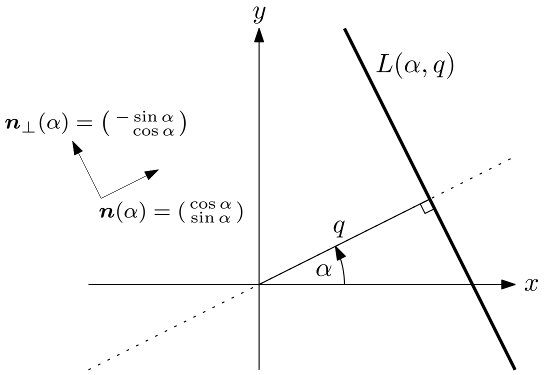

y
$L(\alpha, q)$
$\boldsymbol{n}_\perp(\alpha) = \left(\begin{smallmatrix} -\sin\alpha \\ \cos\alpha \end{smallmatrix}\right)$
$\boldsymbol{n}(\alpha) = \left(\begin{smallmatrix} \cos\alpha \\ \sin\alpha \end{smallmatrix}\right)$
q
α
x

Figure 2.2: Illustration of the Radon transform (Definition 2.2)

by Feldkamp, Davis, and Kress, and *Fourier reconstruction* all share the idea of first analytically inverting the forward problem before using the discretely measured data. Discussing them in detail, however, is out of the scope of this thesis and the interested reader is referred to Natterer [115, Sec. V.1] or Herman [65] for details on the the methods presented in this and the following sections. Natterer and Wübbeling [116] contains hints on how to implement reconstruction algorithms numerically.

2.2.1 Radon's inversion

For the purpose of summarizing analytic reconstruction methods, we will restrict ourselves to 2D reconstruction. The projections involved can then be formalized using

Definition 2.2 (Radon Transform in 2D)
Let $\mu\colon \mathbf{R}^2 \to \mathbf{R}$ *be a function (a planar image) with suitable smoothness properties and let*

$$\begin{aligned} L(\alpha, q) &= \left\{\boldsymbol{X} \in \mathbf{R}^2 \colon \left\langle \boldsymbol{X}, \boldsymbol{n}(\alpha) \right\rangle = q \right\} \\ &= \left\{\boldsymbol{X} \in \mathbf{R}^2 \colon \boldsymbol{X} = q \cdot \boldsymbol{n}(\alpha) + s \cdot \boldsymbol{n}_\perp(\alpha), s \in \mathbf{R} \right\} \end{aligned}$$

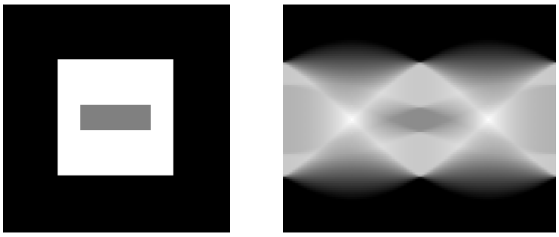
be a line in the plane, parametrized by the angle α *its normal* $\boldsymbol{n}(\alpha) = \left(\begin{smallmatrix} \cos\alpha \\ \sin\alpha \end{smallmatrix}\right)$ *forms with the* x *axis and its distance* q *to the origin (see Figure 2.2). The* Radon transform $\mathcal{R}$ *of* μ *along the line* $L(\alpha, q)$ *is then defined as*

$$(\mathcal{R}\mu)(L) = \int_L \mu(\boldsymbol{X}) \ \mathrm{d}\boldsymbol{X} \tag{2.5a}$$

$$= (\mathcal{R}\mu)(\alpha, q) = \int_{-\infty}^{\infty} \mu\big(q \cdot \boldsymbol{n}(\alpha) + s \cdot \boldsymbol{n}_\perp(\alpha)\big) \ \mathrm{d}s \quad . \tag{2.5b}$$

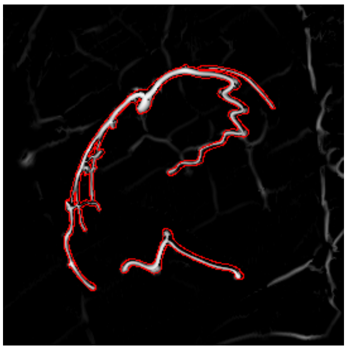
Since $L(\alpha, q) = L(\pi + \alpha, -q)$, the Radon transform has the symmetry property

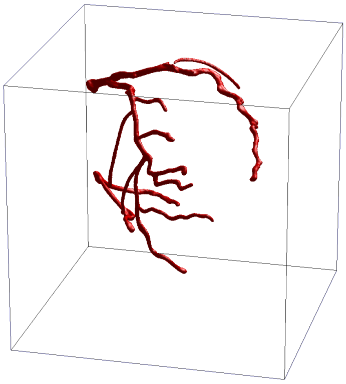
$$(\mathcal{R}\mu)(\alpha, q) = (\mathcal{R}\mu)(\pi + \alpha, -q) \quad .$$



(a) Original image (b) Sinogram data

Figure 2.3: Radon transform of a synthetic image.





Remark. The data emanating from multiple such transformations $(x, y) \mapsto (\alpha, q)$ is often called *sinogram* because single points get mapped to sine waves in (α, d) space. Superimposition of several objects in the original space therefore appear as blurred sinusoidal structures in Radon space (see Figure 2.3). The Radon transform is equivalent to the classical *Hough transform* (invented for the analysis of bubble chamber photographs by Hough [68] in 1959) in its modified form using the angle/offset parametrization and given by Duda and Hart [39].

Note that the Radon transform is an integration over hyperplanes (i. e. over lines in 2D and over planes in 3D). Therefore it can only be used for modeling X-ray projections along lines in a plane. In the case of 3D reconstruction, the data has to be re-sorted to reduce the problem to a series of 2D reconstructions if possible (e. g. in spiral/helical CT). Another possibility is to use the *ray transform* of Definition 2.1. This transform always describes projections along lines (no matter what the image dimension is). It is especially needed in the case of cone beam reconstruction as the data cannot be re-ordered into slices anymore.

The inverse problem of computing the image function μ at a location $\boldsymbol{X}$ from its Radon transform $\mathcal{R}\mu$ can be solved using the definition of the mean value

$$F_{\boldsymbol{X}}(r) := \frac{1}{2\pi} \int_{0}^{2\pi} (\mathcal{R}\mu)\Big(\beta, \langle \boldsymbol{X}, \boldsymbol{n}(\beta) \rangle + r\Big) \, \mathrm{d}\beta \tag{2.6}$$

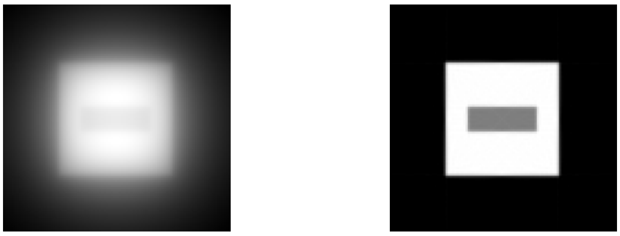
of the line integrals of all tangents to the circle with radius r centered at the fixed location $\boldsymbol{X}$.

Theorem 2.3 (Inversion of the Radon transform)
An inversion of the Radon transform (2.5b) *can be achieved by*

$$\mu(\boldsymbol{X}) = -\frac{1}{\pi} \, \mathrm{p.\,v.} \int_{0}^{\infty} \frac{1}{r} \, \mathrm{d}F_{\boldsymbol{X}}(r) = -\frac{1}{\pi} \, \mathrm{p.\,v.} \int_{0}^{\infty} \frac{1}{r} F'_{\boldsymbol{X}}(r) \, \mathrm{d}r \quad ,$$

where $F_{\boldsymbol{X}}(r)$ is the mean tangent integral as defined in (2.6)*. The first inversion formula is given using a* Riemann-Stieltjes integral*. This can be expressed in the second formula given enough differentiability of $F_{\boldsymbol{X}}(r)$.*

Proof. See Radon [132] or the error-corrected English translation Radon [133]. ■



(a) Unfiltered backprojection (b) Ramp-filtered backprojection

Figure 2.4: Effect of filtering before backprojection. See Figure 2.3 for the corresponding original and sinogram data.

Using Theorem 2.3 and equation (2.6) with $\mathcal{R}\mu = g$ yields the explicit and analytically exact inversion formula

$$\mu(\boldsymbol{X}) = -\frac{1}{2\pi^2}\,\mathrm{p.\,v.}\int_0^\infty \frac{1}{r}\int_0^{2\pi} \frac{\mathrm{d}g}{\mathrm{d}r}\Big(\beta, \big\langle \boldsymbol{X}, \boldsymbol{n}(\beta)\big\rangle + r\Big)\ \mathrm{d}\beta\ \mathrm{d}r \tag{2.7}$$

for $\mu(\boldsymbol{X})$ using *all* the line integrals g in the plane. Due to the complexity of the inversion formula and the sparsity of the measured line integrals g, this formula is not of direct practical use to CT reconstruction. But it can be used as a starting point for deriving filtered back projection formulas.

2.2.2 Filtered back projection

The *filtered back projection* (FBP) gets its name through the insight gained into Radon's inversion formula when derived in another way (see Herman [65, secs. 6.2 and 15.3]). This derivation is based on the identity

$$\mu = \frac{1}{4\pi}\,\mathcal{R}^*\,\mathcal{H}_q\frac{\mathrm{d}}{\mathrm{d}q}\,(\mathcal{R}\mu)\quad , \tag{2.8}$$

where $\mathcal{R}^*$ is the adjoint operator of the Radon transform[3] as defined in Definition B.1 (a simple back projection) and $\mathcal{H}_q$ is the Hilbert transform (see Definition B.2) w. r. t. the offset parameter q. $\mathcal{H}_q\frac{\mathrm{d}}{\mathrm{d}q}$ is known as *ramp filter* and emphasizes higher frequencies of the projection data before the back projection. This mathematically founded high-pass filter also practically improves the reconstruction result compared to a simple, unfiltered back projection (see Figure 2.4). Substituting $\mathcal{R}\mu = g$ on the right side of (2.8) and using the Definition B.2 of the Hilbert transform turns the identity into the reconstruction formula

$$\mu = -\frac{1}{4\pi^2}\,\mathrm{p.\,v.}\int_{\mathbf{R}} \frac{1}{q}\int_0^{2\pi} \frac{\mathrm{d}}{\mathrm{d}q}g\Big(\alpha, \big\langle \boldsymbol{X}, \boldsymbol{n}(\alpha)\big\rangle + q\Big)\ \mathrm{d}\alpha\ \mathrm{d}q \quad .$$

---

[3] Note that the multiplicative constant in (2.8) depends on how the back projection is exactly defined.

This is identical to Radon's inversion when adjusting integration boundaries (cf. (2.7)).

Another example is based on the identity

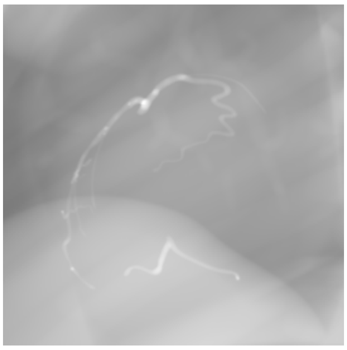
$$(\mathcal{R}^* w_b) * \mu = \mathcal{R}^* \big(w_b * (\mathcal{R}\mu) \big) \quad .$$

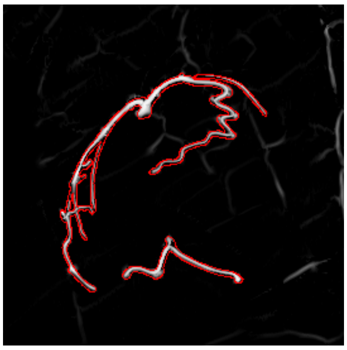
Inserting the measured line integrals g for $\mathcal{R}\mu$ on the equation's right side and choosing w_b so that $\mathcal{R}^* w_b$ approximates the Dirac distribution, one obtains

$$\mu \approx \mathcal{R}^* (w_b * g) \quad .$$

$(\mathcal{R}^* w_b) \approx \delta$ is usually achieved using the frequency domain and Fourier transform to find a well-suited function w_b – depending on the desired visualization result.

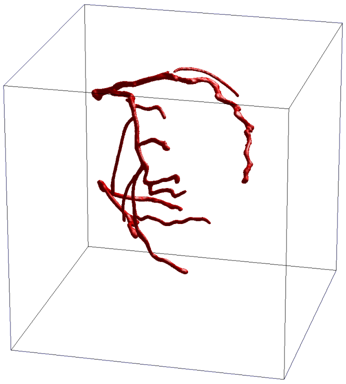
For a good overview of filtered back projection approaches, see Turbell [165]. Practical considerations when implementing FBP are found in Natterer and Wübbeling [116]. FBP for cone beam geometry was tackled by Feldkamp, Davis, and Kress [46] and modified by Wiesent et al. [170] to enable the use of arbitrary projection matrices defining the acquisition geometry.





2.2.3 Fourier reconstruction

A third way of deriving an analytic inversion that should be mentioned is the *Fourier reconstruction*. This approach starts by using the *projection slice theorem*:



Theorem 2.4 (Projection slice theorem in 2D)
The projection slice theorem *(a.k.a.* Fourier slice theorem *or* central slice theorem*) states that the 1-dimensional Fourier transform of projection data (transforming the offset parameter) is equal to a slice through the 2D-Fourier-transformed original data, located at the origin and orthogonal to the projection direction. This is expressed by the identity*

$$(\mathcal{F}_q \mathcal{R}\mu)(\alpha, \rho) = \mathcal{F}_{\boldsymbol{X}} \mu\big(\rho \cdot \boldsymbol{n}(\alpha)\big) \quad . \tag{2.9}$$

On the right side of this equation, $\mathcal{F}_{\boldsymbol{X}}$ denotes the 2-dimensional Fourier transform w. r. t. the Cartesian coordinates $\boldsymbol{X}$ whereas the Fourier transform on the left side is applied in the offset parameter q only (yielding the frequency coordinate ρ).

Remark. Since there are several possible definitions for the Fourier transform, we fix the one given in Definition B.3.

Applying the inverse 2D Fourier transform to (2.9) and inserting the measurements yields the explicit reconstruction

$$\mu\big(\rho \cdot \boldsymbol{n}(\alpha)\big) = \mathcal{F}_{\boldsymbol{X}}^{-1}\big((\mathcal{F}_q g)(\alpha, \rho)\big) \quad . \tag{2.10}$$

Fourier reconstruction is very efficient due to the availability of the Fast Fourier Transform (FFT) but using (2.10) without further modifications results in severe artifacts. *With* such modifications, Fourier reconstruction is comparable in speed and quality to filtered back projection.

Shepp and Logan [154] perform some comparisons between analytic (in this case the Fourier reconstruction) and iterative reconstruction methods (in addition to introducing the widely adopted *Shepp-Logan phantom*).

The governing thought for all the reconstruction algorithms presented so far is that it is possible to find analytical and explicit inversion formulas that are then discretized in the last step before applying them in a real setting. This results in explicit computations that reconstruct the unknown image μ in a single iteration. The drawback is that the rather strict assumptions are usually not met by the data measurements, necessitating all kinds of corrections for the data "defects" and preventing a straightforward application to irregularly sampled or incomplete data.

2.3 Iterative Methods

In contrast to the analytic methods described in the preceding section, iterative methods are based on a prior discretization and subsequent inversion of (2.4). With the availability of relatively low-cost computational power, it seemed practical to solve reconstruction problems numerically. With this focus on numerical computation, it makes sense to first model the unknowns and data measurements as they are represented in a computer on a discrete grid:

Let

$$g_i := g(L_i) \quad , \quad 1 \leq i \leq m$$

be the measurements and

$$\mu_j = \mu(\boldsymbol{X}_j) \quad , \quad 1 \leq j \leq m$$

the unknown attenuation coefficients. The vector $\boldsymbol{\mu}$ of all coefficients is the linearized voxel volume (or pixel plane if working in 2D) that originates from the discretization of the reconstruction domain on a rectangular grid with n voxels with coordinates $\boldsymbol{X}_j$, $1 \leq j \leq n$.

Remark. Note that we have used the same symbols for the discretized variables as for the continuous ones before. This duplicate usage is not confusing if the reader considers that everything is discrete for the remainder of this section.

With these definitions, (2.4) translates to

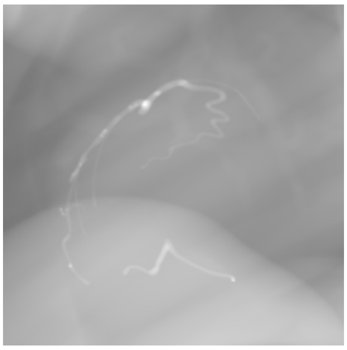
$$\sum_{j=1}^{n} a_{ij}\mu_j = g_i \quad \forall\, 1 \leq i \leq m \quad ,$$

where a_{ij} is the contribution of voxel μ_j to the i^{th} line integral. Stacking all equations and collecting the weights in a matrix $\boldsymbol{A} \in \mathbf{R}^{m \times n}$ results in the linear system of equations

$$\boxed{\boldsymbol{A}\boldsymbol{\mu} = \boldsymbol{g} \ .} \tag{2.11}$$

In order to have an (over-)determined system of equations, $m >= n$ and $\operatorname{rk} \boldsymbol{A} = n$ are required. This is achieved by choosing the resolution of the discretization so that there are less unknowns than measurements.

The values of the weights in the *system matrix* $\boldsymbol{A}$ depend not only on the projection geometry but also on the way the image domain is discretized and how



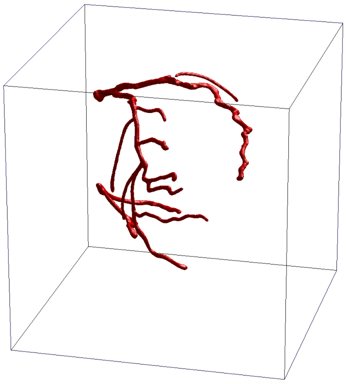
the line integrals are approximated by summation. Advanced discretizations (e. g. using Kaiser-Bessel basis functions, a.k.a. *blobs*) will not be discussed here and we will assume constant values on each rectangular image element. See Turbell [165, ch. 5] for an overview of ways to discretize line integrals. The translation of integration to discrete summation will be subject of further discussions in the following sections. Note, however, that the system matrix $\boldsymbol{A}$ can be used to model all kinds of effects, not just a simple integration.

There is a wide variety of methods available for the solution of systems of over-determined linear equations like (2.11). The goal, in fact, is to compute the optimal solution in a least squares sense, i. e.

$$\boldsymbol{\mu} = \arg\min_{\boldsymbol{\mu}} \|\boldsymbol{A}\boldsymbol{\mu} - \boldsymbol{g}\| \quad . \tag{2.12}$$

One approach is trying to solve the normal equation $\boldsymbol{A}^\mathsf{T}\boldsymbol{A}\boldsymbol{\mu} = \boldsymbol{A}^\mathsf{T}\boldsymbol{g}$. But it is generally a bad idea to square $\boldsymbol{A}$'s condition number that way. Another solution would be to factorize $\boldsymbol{A}$ using a QR decomposition and obtain $\boldsymbol{\mu} = \boldsymbol{R}_1^{-1}\boldsymbol{Q}_1^\mathsf{T}\boldsymbol{g}$. However, most of these methods are not applicable to our setting, regarding the size of the problem: As of 2010, newer C-arm systems typically produce about 200 projection images with 1000 px × 1000 px each, resulting in approximately $m =$ 2tighttimes10^8 measurements. The number of unknowns in the attenuation volume then is in a similar range, e. g. $n \approx$ 1.3tighttimes10^8 for a reconstruction volume of $(500\,\mathrm{vx})^3$. Thus, the matrix $\boldsymbol{A}$ is of the size "hundreds of millions, squared" and requires a fast, iterative solver.

Remark. See e. g. Herman [65, ch. 11] for details on the the methods presented in this and the following sections and Mueller [114] for comments and the application of iterative methods to cone beam CT.



2.3.1 Kaczmarz' Method for Solving Systems of Linear Equations

A suitable method to solve (2.11) was designed by Kaczmarz long before medical image reconstruction was an application. In fact, his iterative method is well-suited and more than just competitive (at least in its newer variations) for solving large unstructured systems of linear equations or non-linear least squares problems.

Kaczmarz' original method [82] cycles through the lines of (2.11), incrementing the line number with the iteration index k and starting over with the first line after m iterations. For each line r, the vector $\boldsymbol{\mu}$ gets projected onto the hyperplane $\langle \boldsymbol{a}_r, \boldsymbol{\mu}(k)\rangle = g_r$ with $\boldsymbol{a}_r^\mathsf{T}$ being the r^{th} row of $\boldsymbol{A}$ (see Figure 2.5). This can also be interpreted as a gradient descent for (2.12) if only single lines of $\boldsymbol{A}$ are taken into account at a time. Introducing a relaxation parameter $\lambda(k)$, the algorithm can formally be written as

$$\boldsymbol{\mu}(k+1) = \boldsymbol{\mu}(k) + \lambda(k)\frac{g_{r(k)} - \langle \boldsymbol{a}_{r(k)}, \boldsymbol{\mu}(k)\rangle}{\|\boldsymbol{a}_{r(k)}\|^2}\boldsymbol{a}_{r(k)} \tag{2.13}$$

with $r(k) = (k \bmod m) + 1$. $\boldsymbol{\mu}(0)$ is usually initialized with zeros, guaranteeing convergence if the relaxation parameter is chosen so that $0 < \lambda(k) < 2$. For overdetermined systems of equations (as is usually the case in CT), Kaczmarz' method converges to the minimizer of $\sum_{r=1}^{n}\left(\frac{g_r - \langle \boldsymbol{a}_r, \boldsymbol{\mu}\rangle}{\|\boldsymbol{a}_r\|}\right)^2$ if $\lambda(k) \xrightarrow{k\to\infty} 0$.

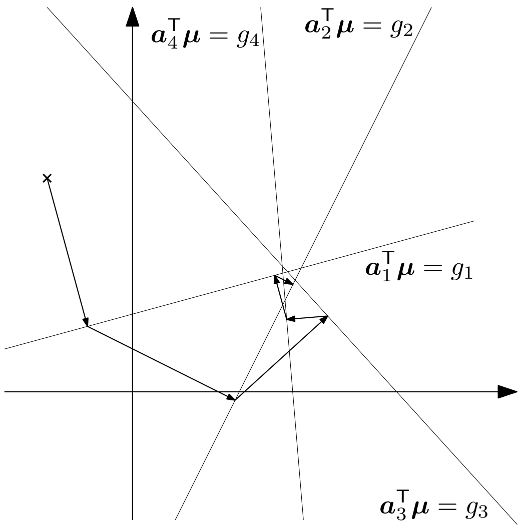

$\boldsymbol{a}_4^\mathsf{T}\boldsymbol{\mu} = g_4$
$\boldsymbol{a}_2^\mathsf{T}\boldsymbol{\mu} = g_2$
$\boldsymbol{a}_1^\mathsf{T}\boldsymbol{\mu} = g_1$
$\boldsymbol{a}_3^\mathsf{T}\boldsymbol{\mu} = g_3$

Figure 2.5: Illustration of Kaczmarz' method in 2D with $\lambda(k) = 1$

2.3.2 Algebraic Reconstruction Techique (ART)

Gordon et al. [57] introduced Kaczmarz' algorithm for image reconstruction and coined the name *algebraic reconstruction technique (ART)*, apparently being unaware of the relation of their work to Kaczmarz' at that time and independently developing an additive as well as a multiplicative version. Their former, additive approach for an iterative solution of (2.11) is very similar to Kaczmarz' update equation (2.13) with $\lambda(k) = 1$. The main drawback of ART as proposed in Gordon et al. [57] is the binary assignment of weights to the system matrix $\boldsymbol{A}$ in the hit-or-miss fashion

$$a_{ij} = \begin{cases} 1 & : L_i \text{ intersects voxel } j \\ 0 & : L_i \text{ does not intersect voxel } j \end{cases} .$$

ART exhibits a quite fast convergence but also strong salt'n'pepper artifacts in the reconstruction. This is due to the clod hopping construction of $\boldsymbol{A}$ and can be easily overcome by simple interpolation in the reconstruction volume.

2.3.3 Simultaneous ART (SART)

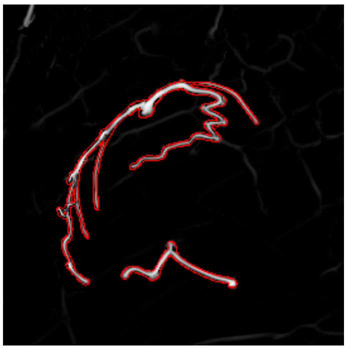
Simultaneous ART was designed with the goal to reduce artifacts that result when applying the ART as defined by Gordon et al. [57]. Published by Andersen and Kak [5], SART mainly differs from ART in three aspects:

- In SART, the forward projection $\langle \boldsymbol{a}_{r(k)}, \boldsymbol{\mu}(k) \rangle$ is no longer performed using binary weights in $\boldsymbol{A}$ but by raycasting with linear interpolation.
- The system matrix' weights computed during the forward projection by linear interpolation are also used for weighting the updates.
- The updates are first computed for all pixels of a common projection direction and then applied at once.

The last change does not seem as important for reducing noise in the reconstruction and newer research actually suggests that random ordering of the updates can be even more effective. The modified calculation of the system matrix $\boldsymbol{A}$ is the crucial improvement that resulted in better algebraic reconstructions.

Comparing SART and the more commonly used filtered back projection algorithms, Chlewicki et al. [28] conclude that SART has more potential in the long run, esp. regarding limited data and the desire for dose reduction with constantly increasing computational power.

2.3.4 Randomized ART

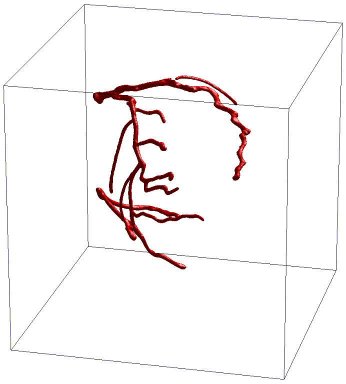


Observing that an ART-type algorithm will converge faster the more orthogonal the hyperplanes of successive iterations, randomized versions have been proposed and successfully tested repeatedly in the literature. But only recently has the convergence rate of such algorithms been quantified in some respect. Strohmer and Vershynin [161] propose performing the Kaczmarz algorithm as given in (2.13) but with a modified selection of the projection lines according to the following random process:[4]

In each iteration, an equation r is chosen with a probability proportional to $\|\boldsymbol{a}_r\|^2$. This can be formalized using the random variable R with the distribution

$$P(R = r) = \frac{\|\boldsymbol{a}_r\|^2}{\sum_{i=1}^{m} \|\boldsymbol{a}_i\|^2} \quad .$$

Choosing $r(k)$ in (2.13) as a realization of the random variable R yields an expected exponential convergence according to Strohmer and Vershynin [161].



2.4 Other Tomographic Reconstruction Methods

In addition to the most widespread reconstruction algorithms for X-ray tomography presented in the preceding sections, there are some other reconstruction methods: In PET and SPECT, a probabilistic approach modeling the Poisson-like distribution of annihilation events leads to the *Maximum Likelihood Expectation Maximization* (EM) reconstruction. This method was proposed in its general form by Dempster et al. [37] and subsequently applied by Shepp and Vardi [155] for emission tomography. *Discrete tomography* is a special case of tomography where only projections along the lattice directions of a rectangular grid with integral or binary attenuation coefficients (as opposed to density functions) are considered. See e. g. Herman and Kuba [66], Natterer and Wübbeling [116, sec. 6.3], and Gardner [52]. The latter reference is a textbook for *geometric tomography* which is concerned with the reconstruction of geometric objects from projections.

---

[4] In spite of the controversy (that can be followed up in vol. 15, pp. 431–440 of the Journal of Fourier Analysis and Applications) between the group of the “ART co-inventor” Gordon and the authors of the presented randomized version, the latter work deserves to be mentioned as it establishes results on expected convergence rates.

2.5 Existing Approaches in Cardiac Cone Beam CT

In this section, we will review approaches developed for the reconstruction of coronary artery trees. The vast literature on motion-compensated reconstruction for other anatomy (like the abdomen) as well as on helical CT[5] or biplane reconstruction will not be evaluated due to the inherent simplifications resulting from the limited motion and/or increased temporal resolution. The same goes for ventricular reconstruction due to its simpler shape and reduced topological complexity as well as for triangulation from a limited number of projections based on manual interaction (e. g., the selection of corresponding feature points in two or more views).

ECG gating and standard reconstruction from complete data
Lauritsch et al. [98] introduce a multi sweep protocol of 3–6 forward and backward runs of RA. This enables an ECG-gated reconstruction from enough data without motion compensation. The temporal resolution depends on the number of sweeps which equals the number of gates that can be chosen without reducing the amount of data per gate. Heart rate variations decrease this temporal resolution since larger gating windows have to be chosen in order to obtain complete sets of projection images.

ECG-gated reconstruction from limited data
Movassaghi et al. [113] optimize (i. e., prolongate) the acquisition protocol in order to improve retrospectively gated FDK reconstructions from at least 10–12 heart cycles during a single RA. Hansis et al. [60, 61] propose to perform ECG-gated, iterative reconstructions specialized to cardiac C-arm CT by incorporating prior knowledge like the sparseness and “tubeness” of the coronary artery tree.

ECG gating and triangulation of coronary model
Movassaghi et al. [111] perform a semi-automatic centerline extraction in projections of a reference phase in combination with epipolar constraints to triangulate a centerline 3D model. This model is then also shaped using automatic vessel extraction and measurements in the additional views. Jandt et al. [76, 77] perform region growing and fast marching, resp., in 3D based on the vesselness measures of all projections of a volumetric point in the same cardiac phase. In [75] such 3D centerline reconstructions are then combined by the authors to obtain a 4D motion vector field using projections from all phases.

4D reconstruction with known motion
An a priori known motion vector field (MVF) greatly simplifies the 4D reconstruction problem. Although not feasible as a stand-alone solution, motion-compensating reconstruction algorithms can be part of other solutions to the full problem, assuming that the heart motion is perfectly periodic and estimated in a prior step. To this end, Schäfer et al. [141, 142] and Prümmer et al. [129] developed motion-compensated FDK-like reconstruction algorithms and Isola et al. [73] an iterative one using a known motion vector field. Blondel et al. [11, 12] also present an ART-type, motion-compensated reconstruction

---

[5] Kachelriess and Kalender [80], Kachelriess et al. [81], and Grass et al. [58] are just a few examples for early work on helical CT of cardiac anatomy using retrospective ECG gating.

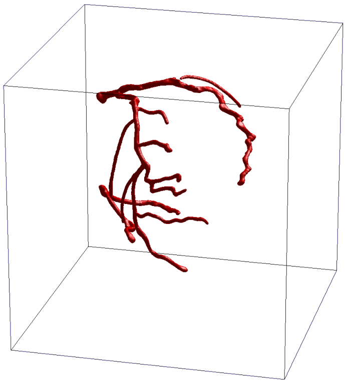
(using simulated DSA projections by creating masks from extracted vessels).

Retrospective ECG gating, triangulation, and 2D motion compensation

Movassaghi et al. [112] rely on ECG gating, manual feature point selection, and epipolar geometry to triangulate a 3D vessel model, then perform a motion-compensation in 2D projection space, and finally compute a 3D reconstruction from the deformed projection images. Hansis et al. [62] also apply the motion correction in the projection space but compute the 2D warping functions from a 2D-alignment of the vessel-enhanced input projections and the forward projected coronary tree extracted from a gated reference reconstruction. Both approaches have the drawback that a motion compensation in projection space has no equivalence in reality. X-ray intensities do not have a physical meaning anymore after a 2D warping.

Retrospective ECG gating, triangulation, and 3D motion compensation

Blondel et al. [10] perform an automatic extraction of the coronaries' centerlines in several key frames and then reconstruct a 3D centerline model in a reference phase while optimizing the extrinsic camera parameters in order to compensate for the breathing motion (assuming it occurs along the body's main axis). Finally, a deformable motion is estimated by fitting the reference reconstruction to the remaining projections and this motion field is then used to compute a motion-compensated, ART-like tomographic reconstruction. In their advanced work, Blondel et al. [13] extract the cardiac phase from the high frequency part of the vertical motion detectable in the projection images. After vessel extraction, a multi-ocular matching for a reference phase yields a 3D reconstruction which is subsequently again fitted to the projection data of the remaining cardiac phases via motion optimization. The obtained motion is again used for a motion-compensated tomographic reconstruction.



Subsequent reconstruction and motion estimation

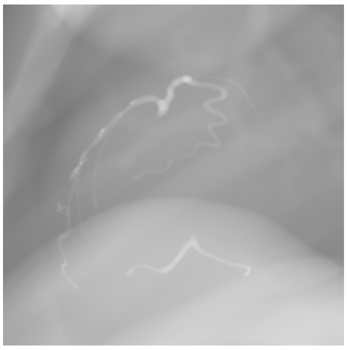
Prümmer et al. [129, 130, 131] and Rohkohl et al. [134] compute a series of retrospectively gated FDK reconstructions and then estimate a 4D motion by non-rigid 3D-3D registration to be able to finally obtain a motion-compensated FDK reconstruction. Using multiple sweeps for obtaining enough data for several gated FDK reconstructions entails loosing some of the contrast agent, dose, and time benefits of the rotational angiographic acquisition protocol.

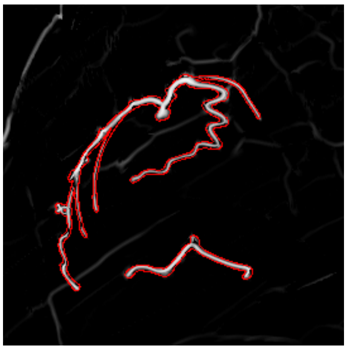
Interleaved reconstruction and motion estimation

Assuming that the motion-compensated FDK reconstruction is a direct and unique solution and can therefore be interpreted as a function of the motion parameters, Rohkohl et al. [136, 137] optimize these parameters w. r. t. an objective function measuring the dissimilarity in 3D and 2D, resp.

Simultaneous reconstruction and motion estimation

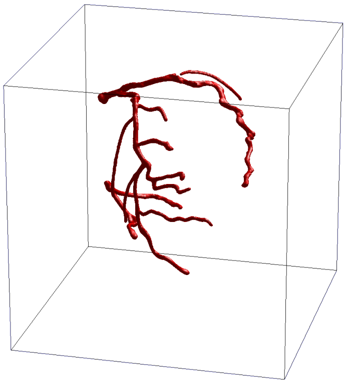
Schomberg [144] and Hansis et al. [63] perform a simultaneous reconstruction of the attenuation volume and the motion using a Kaczmarz/ART-type algorithm. This simultaneous approach better models the physical process of imaging a beating heart but may lack enough cues for motion estimation, esp. from the thin distal parts of the coronaries.

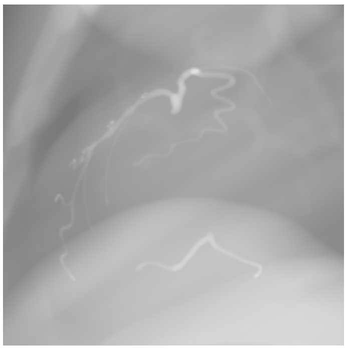




Part II

Methods





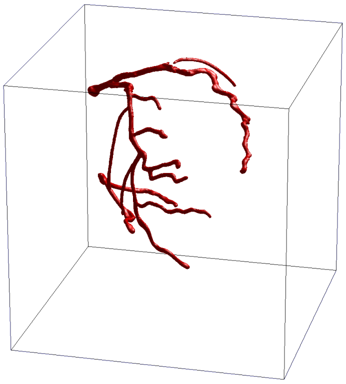
Chapter 3

General Idea

The following few chapters present a novel approach to dynamic shape reconstruction from non-synchronized, probabilistic projections. Before developing the necessary mathematical formulations, we will first describe the problem of cardiac cone beam reconstruction in computer vision terms and lay out the reasoning behind our choices of methods.

3.1 Symbolic Reconstruction

A central assumption underlying this work is that a direct tomographic reconstruction from cardiac cone beam data is not feasible due to the large number of degrees of freedom that are sought. It is also evident from the prior work in this area (see Section 2.5) that there is no consensus on the most promising approach toward cardiac cone beam reconstruction yet. Both, symbolic[1] as well as tomographic approaches have been developed with neither of them being fully convincing.



Given this situation, we opted to develop a symbolic reconstruction that is robust and versatile, enabling a later integration with a tomographic reconstruction (see Section 9.2). Such a symbolic reconstruction problem can be classified using various terminology from the field of computer vision. It can be described as

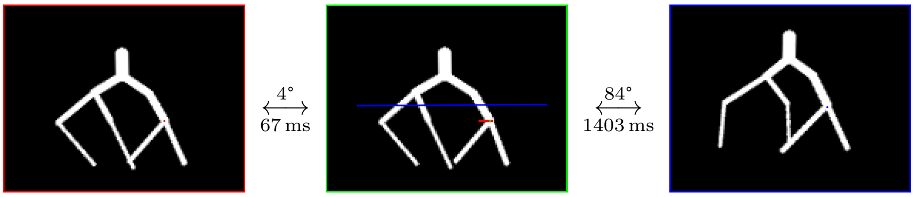
- the recovery of data from indirect measurements, i. e., and inverse problem,
- shape from probabilistic silhouette,
- single view reconstruction from a moving (calibrated) camera or multi-view reconstruction from non-synchronous views (both with dynamic and therefore inconsistent scenery),
- dynamic, deformable, indirect segmentation from projections, and
- scene-flow estimation from non-photometric projections.

Negatively expressed, what makes this setting particularly difficult is that

– there is no photometric information available (as in most stereo-vision approaches),

---

[1] *Symbolic reconstruction* in this case denotes the reconstruction of structural (shape) information from vessel-enhanced images as opposed to attenuation reconstruction from the original X-ray images.



(a) Projection #14 (b) Projection #15 (c) Projection #36

Figure 3.1: Schematic illustration of the inconsistency vs. baseline trade-off by showing epipolar lines of the same bifurcation point from temporally and angularly adjacent (cf. (a) and (b)) and distant (cf. (c) and (b)) projections, resp.

- there is only a single, moving camera, and
- the scene is dynamic and its motion is non-rigid.

It is clear from the latter two points that the input projections are inconsistent if the motion is not known. This results in a trade-off when estimating the 3D position from two projections: If projections that are acquired within a short time are chosen for reconstruction, the scene may be assumed to be static and the projections consistent. This facilitates the matching of corresponding points in the images but the short baseline results in a badly conditioned depth estimation problem. This is illustrated by showing Figures 3.1(a)'s epipolar line (in red) in Figure 3.1(b): The line – constrained to meaningful depth values – perfectly hits the corresponding bifurcation point but with a high depth sensitivity of 22 $\frac{\text{mm}}{\text{px}}$. Avoiding this problem by choosing a wider baseline implies a greater temporal distance and therefore greater inconsistency which also makes the matching difficult: Figures 3.1(c)'s epipolar line shown in blue in Figure 3.1(b) allows a better depth estimation but the inconsistent scene makes matching ambiguous.

3.2 Motivation for Using Level Sets

Deriving a new symbolic reconstruction method first requires improved shape models. Based on the arguments given in the preceding section, a set of requirements was deduced:

a) The shape model should be flexible enough to cover any configuration of coronary vasculature despite the anatomical variations among the population.

b) There has to be some kind of *soft matching* in the spatial domain (as opposed to a hard triangulation).

c) The temporal smoothness should also be enforced by using appropriate models.

d) The reconstruction of the shape and motion shall be performed simultaneously because a separation of shape and motion estimation is not feasible ("chicken and egg problem").

Requirements a) and b) let us choose level sets for modeling the shape of the vasculature. Level sets have the two nice properties that they don't require any explicit handling of the topology (as would be needed if the centerlines were modeled using splines, e. g.) and that their values define a kind of *smooth transition* from *inside* to *outside* by encoding the distance to the interface between the two. The other two points (d and c) support dynamic shape models that are to be optimized to match the given projections. Since level sets are an *Eulerian*[2] shape model, a temporally and spatially smooth motion is also easier to achieve (by regularizing the densely defined motion) than if using *Lagrangian*[2] models.

Our approach of reconstructing a dynamic level set can be seen as an intermediary between coronary modeling and tomographic reconstruction (see Maddux et al. [106, Table 2]). Although working on shapes instead of attenuation fields, the optimization variables are still defined on the whole volume of interest, yielding the desired properties mentioned before as well as some additional benefits:

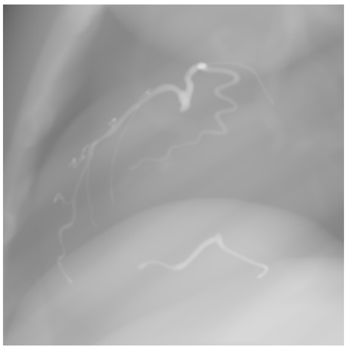
- The volumetric modeling more closely resembles reality because the coronary vasculature is attached to the heart and does not consist of separate vessels.
- The implicitly represented lumen shapes contain more information than centerlines and the distance values simplify measurements (e. g., of the diameter). In addition, this kind of segmentation could be tightly integrated with tomographic methods (see Section 9.2).

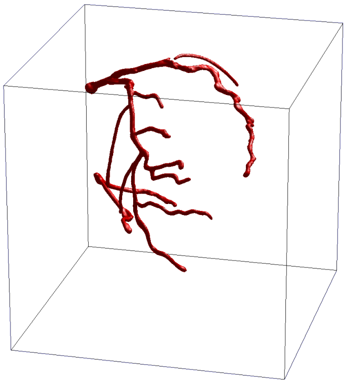
3.3 Related Ideas and Methods

Having chosen level sets for modeling the coronary vasculature, we briefly review other work that is related from a modeling point of view.

Joint segmentation and registration
The idea of an integrated approach for jointly estimating more than one unknown variable was also followed in other settings. The most prominent example is the joint segmentation and registration of two images. This is to say, not an atlas-based segmentation by registering a ground truth image with attached segmentation to a new image, but rather registering two images and jointly segmenting structures of a priori unknown shape. Yezzi, Zöllei, and Kapur [176, 177] emphasize the advantages of combining segmentation and registration into a single energy formulation. However, their work is not related to stereo vision or reconstruction and they focus on rigid and affine motions with only a mention of free form deformations.

Joint segmentation and motion estimation
Other *joint* approaches include the motion segmentation methods developed by Cremers and Schnörr [34], Cremers and Soatto [35], Brox et al. [17], and Schoenemann and Cremers [143] for single-view optical images. In this case,





---

[2] See Chapter 4 for the meaning of *Eulerian* and *Lagrangian*.

an unknown optical flow or scene flow is estimated jointly with a partition of the scene into regions of conforming motion. Cremers [31] uses both, explicit (snakes) as well as implicit (level sets) shape models for realizing this motion segmentation.

Level set reconstruction
In the reconstruction domain, level sets have further been directly used by Yoon et al. [178] for the CT reconstruction of cross sections with a limited number of intensities using multiphase level sets and by Whitaker [168] for the reconstruction of surfaces from range data. Yu and Fessler [179] use level sets to model the boundaries between various anatomical structures in emission tomography and then apply an image gradient penalty that is mostly in effect at the "inner parts" of these structures.

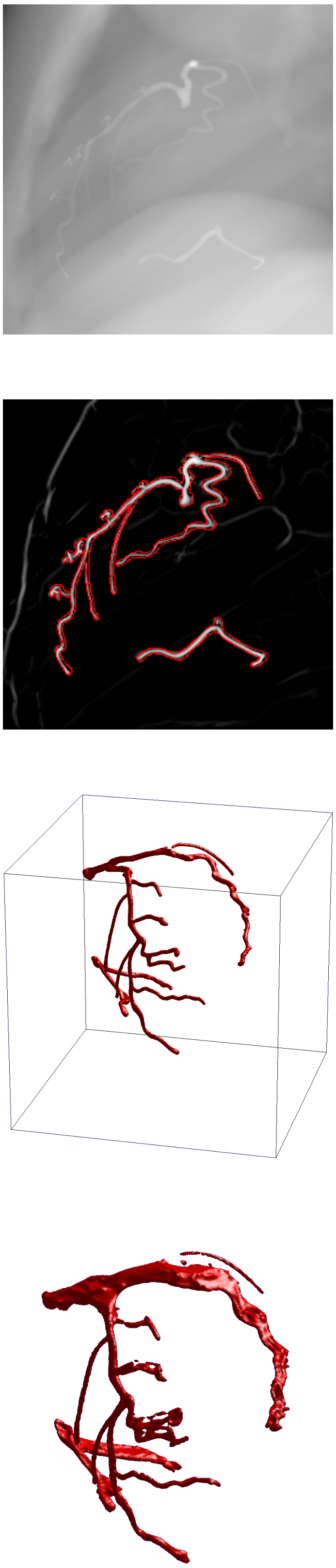
Curves in 3D
Before introducing our shape model of choice, it should be noted that there exist various other ways to represent curved structures like vascular trees in 3D (in addition to splines that are usually employed for centerline modeling): In the level set domain, Ambrosio and Soner [4] and Lorigo et al. [105] represent co-dimension 2 shapes with one level set function (only having non-negative values then) whereas Buchard et al. [18] use two level set functions. Avidan and Shashua [7] and Kaminski et al. [85, 84] give analytic methods for the reconstruction of algebraically representable curves (e. g. motion paths of single points) from projections, incorporating prior knowledge about the topology. Such analytic methods, however, are very sensitive to noise and to deviations from the simplifying path assumptions.

3.4 Components of a Dynamic Level Set Reconstruction

The considerations of the previous sections lead to the development of a new framework for 4D shape reconstruction based on three components (see Figure 3.2), namely

- vessel enhancement in the angiographic projections,
- energy functionals for the level set reconstruction from probabilistic silhouette images (see Chapter 5), and
- dynamic level sets (see Chapter 6).

The methods employed for vessel enhancement are well-known and have been established for many years in the medical image processing domain. This is why they are not covered in this thesis. The other two parts are the core of the presented work and are solved using novel mathematical formulations. These models were developed with the given cardiac reconstruction problem in mind. At the same time, they are derived using very general formulations before assembling the combined framework for our experiments. This ensures a concise description as well as a potential applicability to other reconstruction problems.



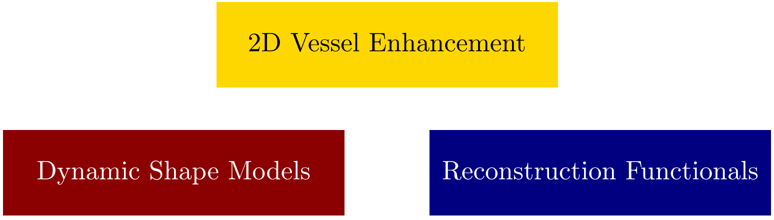
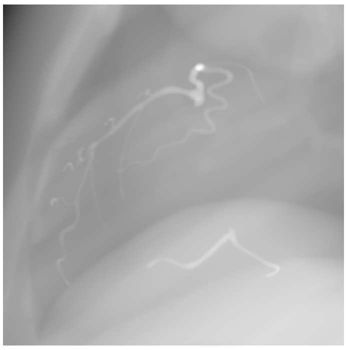
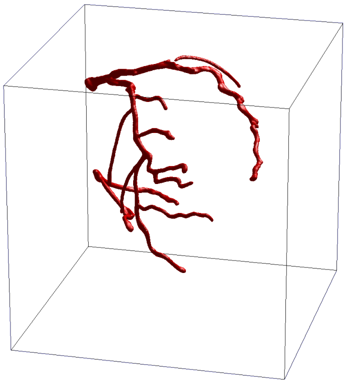

2D Vessel Enhancement
Dynamic Shape Models
Reconstruction Functionals

Figure 3.2: Components of a dynamic symbolic level set reconstruction from angiographic projections

Chapter 4

Active Contours





Before going into the details of our dynamic shape reconstruction components, readers *not* familiar with contour evolution methods (esp. level sets) might find a useful introduction to this topic in this chapter.

Separating structures of interest from the background (as usually sought in segmentation but also in shape reconstruction) can be achieved by modeling the interface between them. If this separation is yet to be found according to some given criteria, one has to start with an initial guess for the interface and then evolve it so that the given criteria are better met. Other terms used for those moving interfaces are *active contours* or *evolving fronts*. Since active contour methods are not a "natural tool" for reconstruction, they are introduced in this chapter.

Remark. See also Li et al. [101] and Chan and Vese [23] for a good classification of level set methods.

4.1 Lagrangian Modeling

Snakes were first introduced by the seminal work of Kass, Witkin, and Terzopoulos [86]. They define energies that, when minimized, drive a contour according to internal (regularizing the shape of the contour) and external (e. g., image-based) forces. Since the solution of the derived Euler-Lagrange equations yields an iterative process moving the contour, the authors coined the term *snake* (due to the similarity of a moving contour in 2D with a moving snake on the ground) for active contours.

More generally, let $\mathcal{H}$ be a hypersurface that is explicitly defined by

$$\boldsymbol{h}\colon \left\{ \begin{array}{rcl} \mathcal{S} \subset \mathbf{R}^{n-1} & \to & \mathbf{R}^n \\ \boldsymbol{s} & \mapsto & \boldsymbol{h}(\boldsymbol{s}) \end{array} \right. ,$$

where n is the dimension of the domain we are working in (usually 2D or 3D) and $\boldsymbol{s}$ is a parameter (which is one-dimensional for curves in 2D and two-dimensional for surfaces in 3D, see Figure 4.1). The set of all points on the curve is then

$$\mathcal{H} = \boldsymbol{h}(\mathcal{S}) \quad .$$

In order to achieve a goal like segmenting an image, one now defines an energy

$$E(\mathcal{H}) = E(\boldsymbol{h}) = \int_{\mathcal{S}} f_{\text{int}}\big(\nabla \boldsymbol{h}(\boldsymbol{s}), \nabla^2 \boldsymbol{h}(\boldsymbol{s})\big) + f_{\text{ext}}\Big(I\big(\boldsymbol{h}(\boldsymbol{s})\big)\Big) \,\mathrm{d}\boldsymbol{s}$$

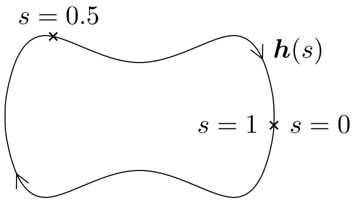


Figure 4.1: Explicitly modeled contour

that depends on an *internal* term f_{int} as well as an *external* term f_{ext}. The internal energy component regularizes the hypersurface so that the result is smooth. It usually contains some combination of first and second derivatives of $\boldsymbol{h}$ and very often includes a measure of the hypersurface's length or area, resp. The other energy component covers influences from external data I, such as images. Before explaining the further steps, we will introduce an example problem and revisit it throughout the rest of this chapter:

Example 4.1
Assuming that one wants to segment a structure located at large gradients of the 2D image I, an appropriate energy term would be

$$E(\boldsymbol{h}) = \int_{\mathcal{S}} \left[1 + g\big(\boldsymbol{h}(s)\big)\right] \cdot \|\boldsymbol{h}'(s)\| \; \mathrm{d}s \tag{4.1}$$

with $\mathcal{S} \subset \mathbf{R}$ and $g\colon \mathbf{R}^2 \to \mathbf{R}$ being a decreasing function of $\|\nabla I\|$, e. g.

$$g(\boldsymbol{x}) = \mathrm{e}^{-\|\nabla I(\boldsymbol{x})\|} \quad .$$

The first addend in functional (4.1) measures the overall length and is therefore a regularizer preventing too much curvature. The second addend is the external energy, trapping the curve on large image gradients.

Remark. Note that the energy (4.1) was defined for an arbitrary parametrization s. Using the arc length l, the same energy can be expressed as

$$E(\boldsymbol{h}) = \int_0^L 1 + g\big(\boldsymbol{h}(l)\big) \; \mathrm{d}l \quad .$$

But this representation can not be derived directly, since non-trivial length-preserving variations do not exist. This is why a parameter transformation from l to s was performed using the relationship $\mathrm{d}l = \|\boldsymbol{h}'(s)\|\,\mathrm{d}s$ (derived from the length formula $l(s) = \int_{s_0}^{s} \|\boldsymbol{h}'(\zeta)\|\,\mathrm{d}\zeta$).

For minimizing the energy E w. r. t. the curve $\boldsymbol{h}$, we compute the variational derivative of E w. r. t. $\boldsymbol{h}$ and use the result as update

$$\frac{\mathrm{d}\boldsymbol{h}}{\mathrm{d}\tau}(\boldsymbol{s}, \tau) = -\delta_{\boldsymbol{h}} E\left(\boldsymbol{h};\ \delta_{\boldsymbol{s}}\right) \tag{4.2}$$

for $\boldsymbol{h}$ (see Appendix D.3 for details) over the artificial iteration time τ.

Remark (Notation for Gâteaux Differentials). Note that a common notation is to write $\frac{\delta E}{\delta \boldsymbol{h}}$ for the integrand $I(\boldsymbol{h})$ of the Gâteaux differential

$$\int_{\mathcal{S}} \langle I(\boldsymbol{h}), \delta\boldsymbol{h}\rangle \;\mathrm{d}\boldsymbol{s} \quad .$$

This integrand could then directly be used as gradient and update for the curve $\boldsymbol{h}$. In this work, however, some directional derivatives will not be linear w. r. t. the direction $\delta\boldsymbol{h}$, preventing this notation. This is why we will mostly use the complete integral denoted by $\delta_{\boldsymbol{h}}E\,(\boldsymbol{h};\ \delta\boldsymbol{h})$ instead. Calculating the integrand at the point $\boldsymbol{s}$ is then achieved by probing this differential with the Dirac distribution $\delta_{\boldsymbol{s}}$:

$$\frac{\delta E}{\delta \boldsymbol{h}}(\boldsymbol{s}) = \delta_{\boldsymbol{h}}E\,(\boldsymbol{h};\ \delta_{\boldsymbol{s}})$$

Remark. Although there are various optimization algorithms that are in many situations much more efficient than a gradient descent, we will not elaborate on these. The main reason is that we will have a hard time even finding something like a gradient for the functionals developed in this thesis, precluding higher order methods. On the other hand, investigating gradients helps in validating an algorithm's effects, analytically as well as numerically. For these reasons we will give the gradients of all functionals.

Example 4.1 (continued)
The derivative of this energy functional w. r. t. the curve can be computed using Corollary D.3 and then Proposition C.3 (for getting rid of the $\delta h_1'$ and $\delta h_2'$ terms), yielding

$$\delta_{\boldsymbol{h}}E\,(\boldsymbol{h};\ \delta\boldsymbol{h}) = \int_{\mathcal{S}} \underbrace{\Big(\big[1+g(\boldsymbol{h})\big]\cdot\kappa + \langle\nabla_{\boldsymbol{x}}g(\boldsymbol{h}),\boldsymbol{n}\rangle\Big)\cdot\|\boldsymbol{h}'\|}_{=:F(\boldsymbol{h})}\cdot\langle\boldsymbol{n},\delta\boldsymbol{h}\rangle\;\mathrm{d}s \tag{4.3}$$

with

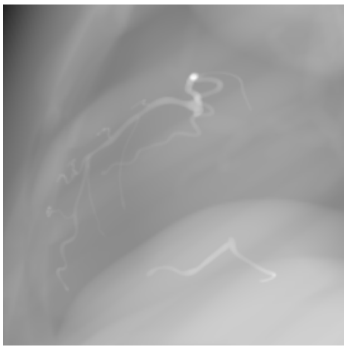
$$\kappa(s) := \frac{h_1'(s)h_2''(s) - h_2'(s)h_1''(s)}{\|\boldsymbol{h}'(s)\|^3} \quad \text{and} \quad \boldsymbol{n}(s) := \frac{\begin{pmatrix} h_2'(s) \\ -h_1'(s)\end{pmatrix}}{\|\boldsymbol{h}'(s)\|}$$

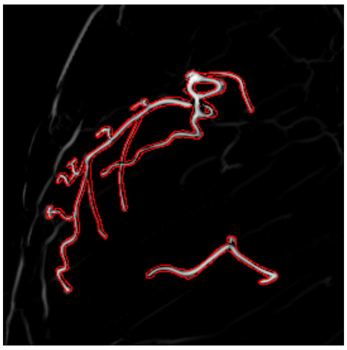
being the curvature (defined so that it is positive for counter-clockwise motion in 2D) and the unit normal to the curve (defined so that it is pointing *outward* for counter-clockwise curves), resp.

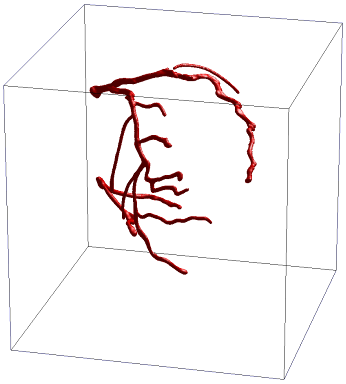
Remark. The differential of the chosen image gradient term $g = \mathrm{e}^{-\|\nabla I\|}$ can be calculated as $\nabla g = -g\cdot \mathrm{H}_I \cdot\nabla I$ (where H_I is the Hessian of the image data), but we will still keep the notation ∇g for brevity and generality.

Putting the energy derivative (4.3) into to steepest descent equation (4.2) yields the update

$$\frac{\mathrm{d}\boldsymbol{h}}{\mathrm{d}\tau} = -\Big(\big[1+g(\boldsymbol{h})\big]\cdot\kappa + \langle\nabla g(\boldsymbol{h}),\boldsymbol{n}\rangle\Big)\cdot\|\boldsymbol{h}'\|\cdot\boldsymbol{n} = -F(\boldsymbol{h})\cdot\boldsymbol{n} \quad . \tag{4.4}$$







This only moves the curve normal to itself (which makes sense because tangential movements would not change the curve's shape and therefore also not change the energy's value). Furthermore, the part $-\big[1+g(\boldsymbol{h})\big]\cdot\kappa$ moves the curve inward (outward) for positive (negative) curvatures κ, smoothing the curve. The speed of this curvature-decreasing motion is higher for smaller image gradients under the curve. The part $-\langle\nabla g(\boldsymbol{h}),\boldsymbol{n}\rangle$ attracts the curve to higher gradients by projecting the appropriate update direction onto the curve's normal. In future references to this example, we will use the shortcut $F(\boldsymbol{h})$, see (4.3). The greater the magnitude of F, the faster will the curve move at the corresponding point.

The abstract update equation (4.2) is not really useful if we do not have any mathematical representation of $\boldsymbol{h}$. In the next step, one has to parametrize the curve in some way in order to be able to algorithmically update its shape. This parametrization can either be done explicitly or implicitly and active contours branch out into *snakes* (Section 4.2) and traditional level set methods (Section 4.3) at this point.

4.2 Snakes

In order to drive the hypersurface $\mathcal{H}$ so that the energy E is minimized, one needs to parameterize the curve $\boldsymbol{h}$. One way to achieve that is to use an explicit representation (like B-splines) with model parameters $\boldsymbol{\alpha}$ so that

$$\boldsymbol{h}\colon \left\{ \begin{array}{ccc} \mathcal{S}\times\mathbf{R}^k & \to & \mathbf{R}^n \\ (\boldsymbol{s};\boldsymbol{\alpha}) & \mapsto & \boldsymbol{h}(\boldsymbol{s};\boldsymbol{\alpha}) \end{array} \right.$$

also depends on the parameter vector $\boldsymbol{\alpha}$, in addition to $\boldsymbol{s}$. Since the final goal is to minimize the energy E w. r. t. the parameters $\boldsymbol{\alpha}$, one is interested in the derivatives

$$\frac{\partial E}{\partial \alpha_i} = \delta_{\boldsymbol{h}} E\left(\boldsymbol{h};\ \frac{\partial \boldsymbol{h}}{\partial \alpha_i}\right)$$

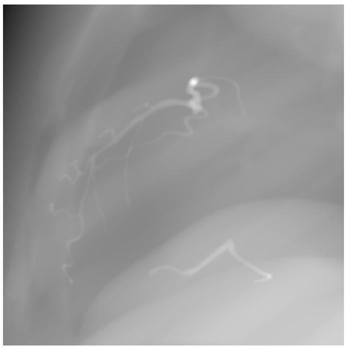
for applying updates

$$\frac{\mathrm{d}\alpha_i}{\mathrm{d}\tau} = -\frac{\partial E}{\partial \alpha_i} = -\delta_{\boldsymbol{h}} E\left(\boldsymbol{h};\ \frac{\partial \boldsymbol{h}}{\partial \alpha_i}\right) \tag{4.5}$$

and therefore evolving $\boldsymbol{h}(\boldsymbol{s},\boldsymbol{\alpha})$ w. r. t. the artificial time τ. Describing and computing the evolution of such an explicitly modeled hypersurface is called *Lagrangian approach.* It has the advantage that there is an explicit way to compute the motion of points on the curve. For evaluating (4.5) one has to sample the integral term along the current hypersurface at a number of points and sum up their contribution to the parameters update.

Example 4.1 (continued)
Assuming that the structure we want to segment is circular, we model the curve as a circle with radius r centered at $\binom{c_x}{c_y}$

$$\boldsymbol{h}\colon \left\{ \begin{array}{ccc} [0,2\pi)\times\mathbf{R}^2\times\mathbf{R}_0^+ & \to & \mathbf{R}^2 \\ \left(s, \begin{pmatrix} c_x \\ c_y \end{pmatrix}, r\right) & \mapsto & \begin{pmatrix} c_x \\ c_y \end{pmatrix} + r\cdot\begin{pmatrix} \cos(s) \\ \sin(s) \end{pmatrix} \end{array} \right. .$$



In this case, c_x, c_y, and r are the only model parameters to be optimized. The respective derivatives are

$$\frac{\partial \boldsymbol{h}}{\partial c_x} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \ , \quad \frac{\partial \boldsymbol{h}}{\partial c_y} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \ , \quad \frac{\partial \boldsymbol{h}}{\partial r} = \begin{pmatrix} \cos(s) \\ \sin(s) \end{pmatrix} \quad .$$

Using the previous result (4.3) as well as $\|\boldsymbol{h}'(s)\| = r$, $\boldsymbol{n}(s) = \begin{pmatrix} \cos(s) \\ \sin(s) \end{pmatrix}$, and $\kappa = \frac{1}{r}$, one obtains the gradient descent

$$\frac{\mathrm{d}c_x}{\mathrm{d}\tau} = -\delta_{\boldsymbol{h}} E\left(\boldsymbol{h};\ \frac{\partial \boldsymbol{h}}{\partial c_x}\right) = -\int_{\mathcal{S}} F(\boldsymbol{h}) \cdot \cos(s)\ \mathrm{d}s$$

$$\frac{\mathrm{d}c_y}{\mathrm{d}\tau} = -\delta_{\boldsymbol{h}} E\left(\boldsymbol{h};\ \frac{\partial \boldsymbol{h}}{\partial c_y}\right) = -\int_{\mathcal{S}} F(\boldsymbol{h}) \cdot \sin(s)\ \mathrm{d}s$$

$$\frac{\mathrm{d}r}{\mathrm{d}\tau} = -\delta_{\boldsymbol{h}} E\left(\boldsymbol{h};\ \frac{\partial \boldsymbol{h}}{\partial r}\right) = -\int_{\mathcal{S}} F(\boldsymbol{h})\ \mathrm{d}s$$

with $F(\boldsymbol{h}) = \left[1 + g(\boldsymbol{h})\right] + r \cdot \left\langle \nabla g(\boldsymbol{h}), \begin{pmatrix} \cos(s) \\ \sin(s) \end{pmatrix} \right\rangle$.

Example 4.1 (continued)

Building a more general model can be achieved using B-splines to obtain

$$\boldsymbol{h}\colon \left\{ \begin{array}{rcl} [s_0, s_1] \times \mathbf{R}^{k \times 2} & \to & \mathbf{R}^2 \\ (s, \boldsymbol{\alpha}) & \mapsto & \sum_{i=1}^{k} \boldsymbol{\alpha}_i N_i(s) \end{array} \right. \quad .$$

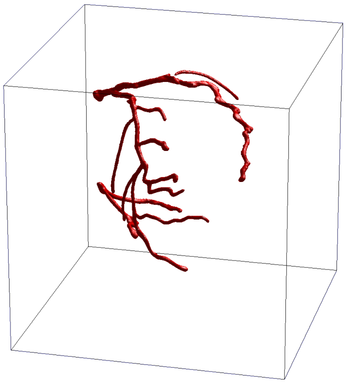
Now the B-spline coefficients $\boldsymbol{\alpha}$ are the parameters to be optimized. The respective derivatives are

$$\frac{\partial \boldsymbol{h}}{\partial \alpha_{i,1}} = \begin{pmatrix} N_i(s) \\ 0 \end{pmatrix} \quad \text{and} \quad \frac{\partial \boldsymbol{h}}{\partial \alpha_{i,2}} = \begin{pmatrix} 0 \\ N_i(s) \end{pmatrix} \quad .$$

The gradient descent is now

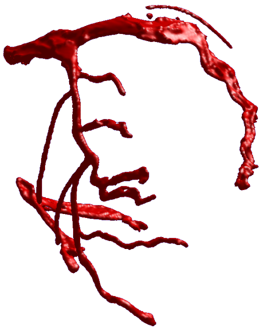
$$\frac{\mathrm{d}\alpha_{i,1}}{\mathrm{d}\tau} = -\delta_{\boldsymbol{h}} E\left(\boldsymbol{h};\ \frac{\partial \boldsymbol{h}}{\partial \alpha_{i,1}}\right) = -\int_{\mathcal{S}} F(\boldsymbol{h}) \cdot \left\langle \boldsymbol{n}, \begin{pmatrix} N_i(s) \\ 0 \end{pmatrix} \right\rangle\ \mathrm{d}s$$

$$\frac{\mathrm{d}\alpha_{i,2}}{\mathrm{d}\tau} = -\delta_{\boldsymbol{h}} E\left(\boldsymbol{h};\ \frac{\partial \boldsymbol{h}}{\partial \alpha_{i,2}}\right) = -\int_{\mathcal{S}} F(\boldsymbol{h}) \cdot \left\langle \boldsymbol{n}, \begin{pmatrix} 0 \\ N_i(s) \end{pmatrix} \right\rangle\ \mathrm{d}s \quad ,$$



or, by assembling both derivatives in one equation (and assuming a component-wise integration symbol)

$$\frac{\mathrm{d}\boldsymbol{\alpha}_i}{\mathrm{d}\tau} = -\delta_{\boldsymbol{h}} E\left(\boldsymbol{h};\ \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\alpha}_i}\right) = -\int_{\mathcal{S}} F(\boldsymbol{h}) \cdot N_i(s) \cdot \boldsymbol{n}\ \mathrm{d}s \quad .$$



Snakes can be powerful in cases where strong geometric prior knowledge can directly be incorporated into the model. However, a snake's topology is fixed and the explicit parametrization requires some extra work during implementation. These two drawbacks can be overcome by translating the results from the Lagrangian modeling into an Eulerian framework as explained in the next section.

4.3 Lagrangian to Eulerian – Classical Level Set Methods

Osher and Sethian [124] re-introduced another way of modeling active contours. They describe evolving fronts as the isolevel contour of an embedding function:

Definition 4.2 (Level set function)
Let $\mathcal{H} \subset \Omega$ be a closed hypersurface or submanifold of $\Omega \subset \mathbf{R}^n$ with codimension 1 *(e. g., a curve in 2D or a surface in 3D). A function*

$$\Phi\colon \begin{cases} \Omega \subset \mathbf{R}^n & \to \quad \mathbf{R} \\ \boldsymbol{x} & \mapsto \quad \Phi(\boldsymbol{x}) \end{cases}$$

with the property

$$\Phi(\boldsymbol{x}) \begin{cases} < 0 & : \boldsymbol{x} \textit{ is } \text{inside } \mathcal{H} \\ = 0 & : \boldsymbol{x} \in \mathcal{H} \\ > 0 & : \boldsymbol{x} \textit{ is } \text{outside } \mathcal{H} \end{cases} \tag{4.6}$$

is called level set function *for the hypersurface $\mathcal{H}$. Possible notations for this implicit definition of the hypersurface are then*

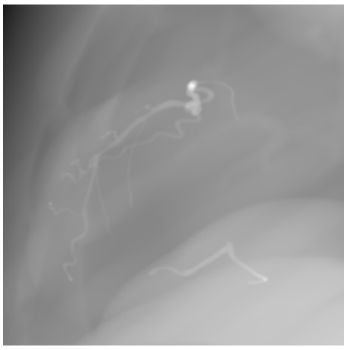
$$\mathcal{H} = \{\Phi = 0\} := \{\boldsymbol{x} \in \Omega\colon \Phi(\boldsymbol{x}) = 0\} = \Phi^{-1}(0) \quad . \tag{4.7}$$

Remark. The choice of the signs in (4.6) as well as the value 0 in Equations (4.6) and (4.7) are arbitrary. However, the zero level set is commonly used since it simplifies the calculus a bit. Our choice of signs for inside and outside result in an outward-pointing gradient of Φ.

A simple example for such a level set function is the signed distance from a given contour. E. g., the zero level set of the function $\Phi(\boldsymbol{x})) = \|\boldsymbol{x}\| - r$ would be the circle with radius r centered at the origin. In practice, Φ is not calculated analytically but discretized on a rectangular grid (see Figure 4.2). The front is then the isosurface of the data set storing Φ. It can be extracted, e. g., using the marching cubes algorithm by Lorensen and Cline [104] or using the more recent method by Kobbelt et al. [91].

The evolution of a front that is modeled this way is now indirectly achieved by modifying the level set function. This also changes its zero level set and therefore the modeled shape (see Figure 4.3). This implicit modeling of an evolving contour by a function defined on the full space is called the *Eulerian approach.* It has the advantage of handling topological changes intrinsically since the front is never really modeled explicitly. Furthermore, the numerical computations can be performed on a fixed rectangular grid. And although this implicit representation seems to be less intuitive, all geometric quantities like the normal

$$\boldsymbol{n} = \frac{\nabla\Phi}{\|\nabla\Phi\|}$$



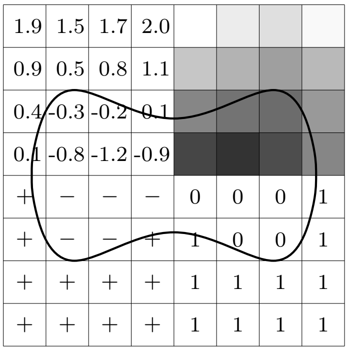
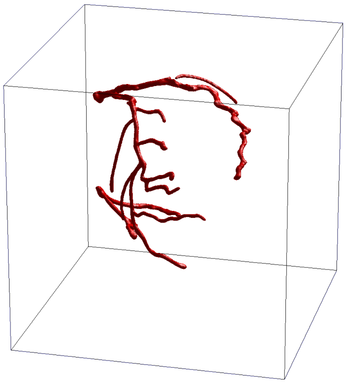

1.9 1.5 1.7 2.0
0.9 0.5 0.8 1.1
0.4 -0.3 -0.2 0.1
0.1 -0.8 -1.2 -0.9
+ − − − 0 0 0 1
+ − − + 1 0 0 1
+ + + + 1 1 1 1
+ + + + 1 1 1 1

Figure 4.2: Representation of a curve by a discretized level set function (visualized using level set function values, colors, signs, and Heaviside values, resp.).



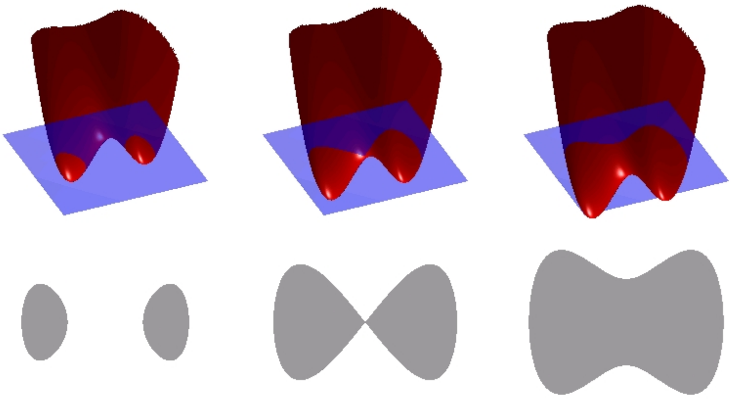


Figure 4.3: An implicitly evolved contour in 2D. (Image source (modified for this work): `http://commons.wikimedia.org/wiki/File:Level_set_method.jpg`. Original author: Oleg Alexandrov. License: Public domain.)

and the curvature

$$\left.\begin{matrix}\kappa\\ 2H\end{matrix}\right\} = \operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right)$$

(where κ is the curvature of a curve in 2D and H is the mean curvature of a surface in 3D, resp.) can be expressed quite easily (see also Appendix E).

Two different ways for finding the evolution equation for the level set function can be distinguished. One simply translates movements of the front computed using the Lagrangian approach from Section 4.2 into the level set formulation while the other never uses an explicit model of the front and directly starts with an energy functional depending on Φ. Both approaches lead to an Eulerian formulation and are described in this and the following section, resp.

Remark. See also Xu et al. [172, 173] on the relationship between explicitly and implicitly modeled active contours.

The more traditional and at first glance less challenging way to construct an update equation for the level set function is to translate a previously computed motion of a front into the level set formalism: Let $\boldsymbol{h}(\boldsymbol{s},\tau)$ be an explicitly modeled curve following an energy-minimizing motion as computed in (4.2). This Lagrangian motion can now be translated into an Eulerian description (see also Sethian [150]). Introducing an artificial iteration time τ in (4.6) yields

$$\Phi\big(\boldsymbol{h}(\boldsymbol{s},\tau),\tau\big) = 0 \quad \forall\, \boldsymbol{s} \in \mathcal{S},\ \forall\, \tau \geq 0$$

which is equivalent to the two equations

$$\Phi\big(\boldsymbol{h}(\boldsymbol{s},0),0\big) = 0 \quad \forall\, \boldsymbol{s} \in \mathcal{S} \tag{4.8a}$$

$$\text{and} \quad \frac{\mathrm{d}\Phi}{\mathrm{d}\tau}\big(\boldsymbol{h}(\boldsymbol{s},\tau),\tau\big) = 0 \quad \forall\, \boldsymbol{s} \in \mathcal{S},\ \forall\, \tau > 0 \ . \tag{4.8b}$$

Expanding the total differential in (4.8b) yields

$$\frac{\partial\Phi}{\partial\tau} + \left\langle \nabla_{\boldsymbol{x}}\Phi, \frac{\mathrm{d}\boldsymbol{h}}{\mathrm{d}\tau} \right\rangle = 0 \quad \forall\, \tau > 0 \ , \tag{4.9}$$

so that the combination of (4.8a) and (4.9) finally yields

$$\Phi(\boldsymbol{x},0) = 0 \quad \forall\, \boldsymbol{x} \in \boldsymbol{h}(\mathcal{S},0) \tag{4.10a}$$

$$\text{and} \quad \frac{\partial\Phi}{\partial\tau}(\boldsymbol{x},\tau) = -\left\langle \nabla\Phi, \frac{\mathrm{d}\boldsymbol{h}}{\mathrm{d}\tau} \right\rangle\Bigg|_{(\boldsymbol{x},\tau)} \quad \forall\, \boldsymbol{x} \in \boldsymbol{h}(\mathcal{S}),\ \forall\, \tau > 0 \ . \tag{4.10b}$$

Since the hypersurface's motion $\frac{\mathrm{d}\boldsymbol{h}}{\mathrm{d}\tau} = -\frac{\delta E}{\delta \boldsymbol{h}}$ usually has the form

$$\frac{\mathrm{d}\boldsymbol{h}}{\mathrm{d}\tau} = -F(\boldsymbol{h}) \cdot \boldsymbol{n} \quad \text{with} \quad \boldsymbol{n} = \frac{\nabla\Phi}{\|\nabla\Phi\|} \ \text{(see Appendix E.1)},$$

the evolution equation (4.10b) simplifies to

$$\frac{\partial\Phi}{\partial\tau} = -\left\langle \nabla\Phi, -F \cdot \frac{\nabla\Phi}{\|\nabla\Phi\|} \right\rangle = F(\boldsymbol{h}) \cdot \|\nabla\Phi\| \ . \tag{4.11}$$

Remark. Even if a precomputed motion would not be normal to the hypersurface at each point, its normal component would still be the only one affecting the energy functional. This is why it is usually suggested that any motion to be translated into the level set formalism should be projected onto the normal first.

The Hamilton-Jacobi equation (4.11) implicitly moves the zero level set in the same way as the contour $\mathcal{H}$. But it was derived for curve points in $\boldsymbol{h}(\boldsymbol{s})$ only. The remaining question is how to compute updates for the full domain Ω of the level set function Φ. The usual solution is to directly use (4.11) on the whole domain of Φ and implement steps to keep the level set function "well-behaved". Another approach is to construct *extension forces* keeping the signed distance property intact during evolution (see Sethian [150, ch. 11] and Adalsteinsson and Sethian [3]).

Example 4.1 (continued)
Assuming that the curve $\boldsymbol{h}$ was parametrized by arc length, $\|\boldsymbol{h}'\| = 1$ holds true for all curve points so that precomputed motion (4.4) can be rewritten using our level set expressions to

$$\frac{\mathrm{d}\boldsymbol{h}}{\mathrm{d}\tau} = -\left([1+g]\cdot \operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right) + \left\langle \nabla g, \frac{\nabla\Phi}{\|\nabla\Phi\|}\right\rangle\right)\cdot \frac{\nabla\Phi}{\|\nabla\Phi\|} = -F\cdot\frac{\nabla\Phi}{\|\nabla\Phi\|}$$

so that the update equation for Φ (4.11) becomes

$$\frac{\partial\Phi}{\partial\tau} = \left([1+g]\cdot \operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right) + \left\langle \nabla g, \frac{\nabla\Phi}{\|\nabla\Phi\|}\right\rangle\right)\cdot\|\nabla\Phi\| = F\cdot\|\nabla\Phi\| \quad . \qquad (4.12)$$

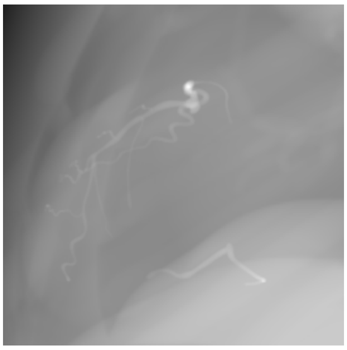
Compared to snakes, the translation into the traditional level set formalism has the following advantages:

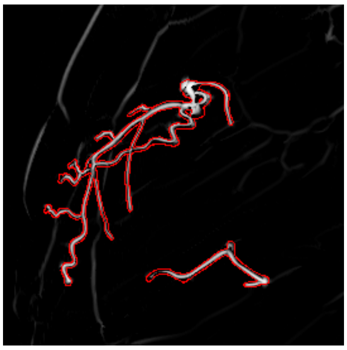
- The data I is sampled on its grid positions (if the level set grid is chosen accordingly) and there is no interpolation or "unnatural" sampling involved.
- The hypersurface $\mathcal{H}$ may change its topology during evolution without the need for special algorithmic treatment.
- Reparametrizations are unnecessary. This is a huge advantage because developing a reparametrization algorithm can be laborious.

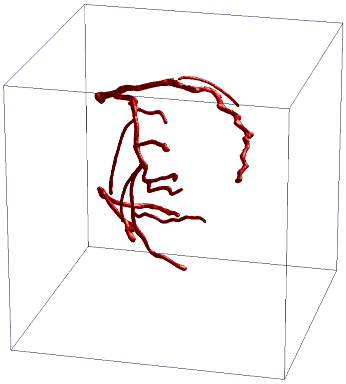
It therefore very often makes sense to translate the computed curve motions (4.2) into the level set formalism. An exception to this rule is the presence of some strong geometric prior knowledge (such as that the final shape is a geometric primitive like a circle). In this case, an explicit modeling may be advantageous compared to additional regularization terms for a level set method.

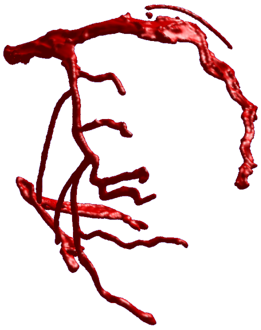
To summarize, the process of deriving a level set evolution from a Lagrangian formulation consisted of the following steps:

1. Derive the equations of motion for $\mathcal{H}$ (e. g., by deriving an energy functional $E(\boldsymbol{h})$ w. r. t. the explicitly modeled front $\boldsymbol{h}$).









2. (Optionally) project the resulting front velocity onto the normal of the curve (since the tangential velocity only affects the parametrization, not the real movement).
3. Insert the normal velocity (usually denoted by F in textbooks, although it is more a velocity than a force) into the level set equation $\frac{\partial \Phi}{\partial \tau} = F\|\nabla\Phi\|$.

4.4 Variational Level Set Methods

Instead of deriving equations of motion for an explicitly modeled front and then translating them into the level set formulation, one may also start right away with an energy functional depending on the level set function (modeling the front) instead of the explicit front:

1. Set up an energy functional $E(\Phi)$, depending on the level set function Φ (implicitly modeling the contour $\boldsymbol{h} = \{\boldsymbol{x}\colon \Phi(\boldsymbol{x}) = 0\}$).
2. Construct the update equation for the level set function by deriving the energy functional: $\frac{\delta E}{\delta \Phi}$

This approach is not only more straightforward due to making a translation step unnecessary. It also introduces additional freedom when modeling the energy functional because one is now able to define energy components for points *not* lying on the front.

Assume that the energy terms are defined as integrals over some function

$$f\colon \left\{ \begin{array}{rcl} \mathbf{R}^n \times \mathbf{R} \times \mathbf{R}^n & \to & \mathbf{R} \\ (\boldsymbol{x}, \Phi, \nabla\Phi) & \mapsto & f(\boldsymbol{x}, \Phi, \nabla\Phi) \end{array} \right. .$$

Remark. Note that the symbols Φ and $\nabla\Phi$ are simply names for the arguments of f. They are not directly related to functions with the same name. Therefore, this definition implies that f only depends on the *values* of $\Phi(\boldsymbol{x})$ and $\nabla\Phi(\boldsymbol{x})$, not on the whole functions. Therefore, f is a *function*, not a functional.

$$f\colon \left\{ \begin{array}{rcl} \mathbf{R}^n \times \mathbf{R} \times \mathbf{R}^n & \to & \mathbf{R} \\ (\boldsymbol{x}, y, \boldsymbol{z}) & \mapsto & f(\boldsymbol{x}, y, \boldsymbol{z}) \end{array} \right. .$$

would have been an equivalent definition. However, the calculus in the remainder of this chapter will be clearer when using the more expressive argument names Φ and $\nabla\Phi$.

Now, let

$$\boxed{E(\Phi) = \int_{\Omega} f\big(\boldsymbol{x}, \Phi(\boldsymbol{x}), \nabla\Phi(\boldsymbol{x})\big)\, \mathrm{d}\boldsymbol{x}} \tag{4.13}$$

be an energy *functional* depending on the function Φ as well as its gradient (which is not mentioned as separate argument on the left-hand side since it results from deriving Φ). The arguments of f with the names Φ and $\nabla\Phi$ are filled with the function values $\Phi(\boldsymbol{x})$ and $\nabla\Phi(\boldsymbol{x})$, resp., at the point $\boldsymbol{x}$.

Remark. Note that the variational level set approach provides even more flexibility in modeling an energy functional. In almost all cases, these functionals have the form (4.13). However, we will also encounter other types of functionals (like the minimum functional) requiring special calculus for computing derivatives in this work.

Integrals over the contour $\mathcal{H}$ can then be expressed using the Dirac distribution (see Theorem E.2) to obtain

$$\int_{\mathcal{H}} f \;\mathrm{dS}(\boldsymbol{\xi}) = \int_{\{\varPhi=0\}} f \;\mathrm{dS}(\boldsymbol{\xi}) = \int_{\varOmega} \delta(\varPhi)\cdot\|\nabla\varPhi\|\cdot f \;\mathrm{d}\boldsymbol{x}$$

whereas integrals over the region inside or outside the contour may be modeled using the Heaviside function as characteristic function to get

$$\int_{\text{inside } \mathcal{H}} f \;\mathrm{d}\boldsymbol{x} = \int_{\{\varPhi<0\}} f \;\mathrm{d}\boldsymbol{x} = \int_{\varOmega} [1 - H(\varPhi)]\cdot f \;\mathrm{d}\boldsymbol{x}$$

$$\int_{\text{outside } \mathcal{H}} f \;\mathrm{d}\boldsymbol{x} = \int_{\{\varPhi>0\}} f \;\mathrm{d}\boldsymbol{x} = \int_{\varOmega} H(\varPhi)\cdot f \;\mathrm{d}\boldsymbol{x} \quad .$$

(using the notation $\{\varPhi \lesseqgtr 0\} := \{\boldsymbol{x}\in\varOmega\colon \varPhi(\boldsymbol{x}) \lesseqgtr 0\}$).

Calculating an update equation (step number 2 above) now involves deriving a functional (E) w. r. t. a function ($\varPhi$) which requires the calculus of variations. When looking for an extremum of the functional $E(\varPhi)$, we need to compute its functional derivative. This functional derivative (called *Gâteaux* or *Fréchet* derivative, depending on the properties of the derivative) has to be calculated w. r. t. a direction, where this direction is a function, an element of an infinite-dimensional space. Using Appendix D.1 yields

$$\begin{aligned}\delta_{\varPhi} E\,(\varPhi;\ \delta\varPhi) = &\int_{\varOmega} f_{\varPhi}\big(\boldsymbol{x},\varPhi(\boldsymbol{x}),\nabla\varPhi(\boldsymbol{x})\big)\cdot\delta\varPhi(\boldsymbol{x})\;\mathrm{d}\boldsymbol{x} + \\ &+ \int_{\varOmega} \big\langle f_{\nabla\varPhi}\big(\boldsymbol{x},\varPhi(\boldsymbol{x}),\nabla\varPhi(\boldsymbol{x})\big), \nabla\delta\varPhi(\boldsymbol{x})\big\rangle\;\mathrm{d}\boldsymbol{x}\end{aligned}$$

for functionals of the form given above. Using integration by parts (see Proposition C.4), the second term can be simplified so that we have $\delta\varPhi$ as a common factor:

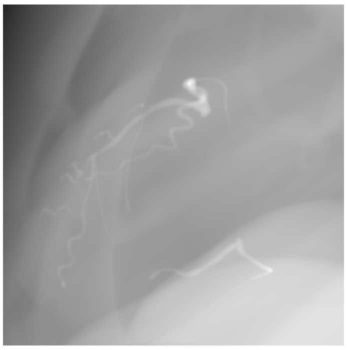
$$\begin{aligned}\delta_{\varPhi} E\,(\varPhi;\ \delta\varPhi) = &\int_{\varOmega} \underbrace{\big[f_{\varPhi}(\varPhi,\nabla\varPhi) - \operatorname{div} f_{\nabla\varPhi}(\varPhi,\nabla\varPhi)\big]}_{=:g(\boldsymbol{x})}\,\delta\varPhi\;\mathrm{d}\boldsymbol{x} + \\ &+ \int_{\partial\varOmega} \big\langle \underbrace{f_{\nabla\varPhi}(\varPhi,\nabla\varPhi),\boldsymbol{\nu}}_{=:h(\boldsymbol{x})}\big\rangle\delta\varPhi\;\mathrm{dS}(\boldsymbol{\xi}) \qquad (4.14)\end{aligned}$$

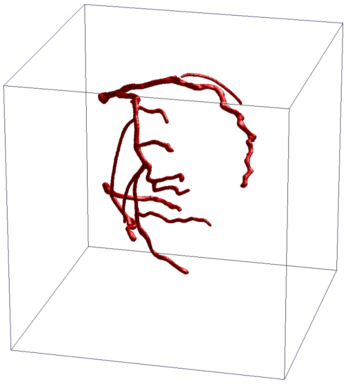
Remark. Terms with derivatives of the level set function are usually introduced when adding regularizers. If f depends on $\varPhi$ only, then $f_{\nabla\varPhi} = 0$, we do not need to integrate by parts, and the calculus of variations becomes a lot easier. However, this is usually not the case since regularization *is* needed.

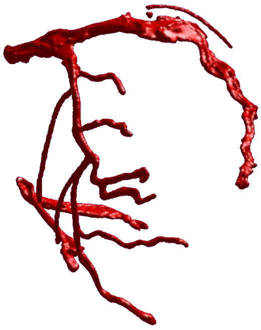
Finally, Appendix D.3 justifies the selection of $-g$ as update for the level set function when iteratively looking for a minimum of E and

$$\boxed{\frac{\partial\varPhi}{\partial\tau}(\boldsymbol{x}) = -g\big(\boldsymbol{x},\varPhi(\boldsymbol{x}),\nabla\varPhi(\boldsymbol{x}),\mathrm{H}_{\varPhi}(\boldsymbol{x})\big) \quad \forall\,\boldsymbol{x}\in\varOmega}$$

is a gradient descent method (assuming that the border integral in (4.14) vanishes).







Remark. The function g usually contains $\delta(\Phi)$ as a multiplicative factor. This is sometimes replaced by $\|\nabla\Phi\|$ in order to obtain a morphological flow, i. e. a flow which is independent of the scaling of Φ. However, this also increases the speed of level sets evolving far away from the zero level set which may or may not be desired.

Example 4.1 (continued)
Directly modeling the same energy as defined in (4.1) in the Eulerian domain can be done using Theorem E.2. The integration over the curve $\{\Phi = 0\}$ is now expressed as

$$E(\Phi) = \int_{\Omega} [1 + g(\boldsymbol{x})] \cdot \delta\big(\Phi(\boldsymbol{x})\big) \cdot \|\nabla\Phi(\boldsymbol{x})\| \;\mathrm{d}\boldsymbol{x} \quad .$$

Computing the variational derivative w. r. t. Φ (using Corollary D.3, Proposition C.1, Corollary C.5, and Proposition C.2) yields

$$\delta_{\Phi} E\,(\Phi;\ \delta\Phi) = \int_{\Omega} \left(-[1+g] \cdot \operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right) - \left\langle \nabla g, \frac{\nabla\Phi}{\|\nabla\Phi\|} \right\rangle \right) \cdot \delta(\Phi) \cdot \delta\Phi \;\mathrm{d}\boldsymbol{x}$$

(under the standard assumption of vanishing boundary integrals) so that the gradient descent update for the level set function (according to Appendix D.3.2) is

$$\frac{\partial\Phi}{\partial\tau} = \left([1+g] \cdot \operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right) + \left\langle \nabla g, \frac{\nabla\Phi}{\|\nabla\Phi\|} \right\rangle \right) \cdot \delta(\Phi) \quad .$$

Optionally replacing the factor $\delta(\Phi)$ with $\|\nabla\Phi\|$ (in order to extend the curve motion to adjacent levels) yields the same update equation as in (4.12). However, another common approach is to use a smooth approximation of the Heaviside function and Dirac functional instead (see E.6).

With the right formulas at hand, a direct modeling is much easier and also more flexible than deducing Lagrangian motions and translating them into the level set framework. Using variational level sets and lifting the restrictions of Lagrangian formulations also broadens the class of energy functions and has some additional advantages:

- The extension of the evolution onto non-hypersurface level sets enables a faster convergence by the creation of new structures not connected to existing ones.
- Integration over the whole domain enables the modeling of region-dependent criteria (like image intensity) instead of just edge-dependent ones (like intensity gradients).

The latter advantage will be demonstrated in a further example:

Example 4.3 (Chan-Vese segmentation [22, 23])
Instead of locating strong gradients in an image (resulting from intensity changes between two different types of tissue), one could directly use the prior information about these tissues' intensities. Given two prior intensity distributions

$$p_{\text{in}}(I) = \mathcal{N}_{\mu_{\text{in}},\sigma}(I) \quad \text{and} \quad p_{\text{out}}(I) = \mathcal{N}_{\mu_{\text{out}},\sigma}(I) \quad ,$$

with different means but the same variance (to keep the example simple), one can associate this information with the conditional probabilities

$$P\big(I(\boldsymbol{x}) = i \mid \Phi(\boldsymbol{x}) < 0\big) = p_{\text{in}}(i) \quad \text{and} \quad P\big(I(\boldsymbol{x}) = i \mid \Phi(\boldsymbol{x}) > 0\big) = p_{\text{out}}(i)$$

for the "inside" ($\Phi < 0$) and "outside" ($\Phi > 0$) regions of a given level set segmentation. The likelihood of the data $I(\boldsymbol{x})$ given the segmentation Φ is then

$$P(I \mid \Phi) = \prod_{\Omega} P\big(I(\boldsymbol{x}) \mid \Phi(\boldsymbol{x})\big)^{\,\mathrm{d}\boldsymbol{x}} = \prod_{\{\Phi<0\}} p_{\text{in}}\big(I(\boldsymbol{x})\big)^{\,\mathrm{d}\boldsymbol{x}} + \prod_{\{\Phi>0\}} p_{\text{out}}\big(I(\boldsymbol{x})\big)^{\,\mathrm{d}\boldsymbol{x}} \ .$$

Using a log-likelihood argument and neglecting constant factors, this leads to the penalizing energy functional

$$\begin{aligned} E(\Phi) &= \int\limits_{\{\Phi<0\}} \big(I(\boldsymbol{x}) - \mu_{\text{in}}\big)^2 \ \mathrm{d}\boldsymbol{x} + \int\limits_{\{\Phi>0\}} \big(I(\boldsymbol{x}) - \mu_{\text{out}}\big)^2 \ \mathrm{d}\boldsymbol{x} \\ &= \int\limits_{\Omega} \big[1 - H(\Phi)\big] \cdot \big(I(\boldsymbol{x}) - \mu_{\text{in}}\big)^2 + H(\Phi) \cdot \big(I(\boldsymbol{x}) - \mu_{\text{out}}\big)^2 \ \mathrm{d}\boldsymbol{x} \end{aligned}$$

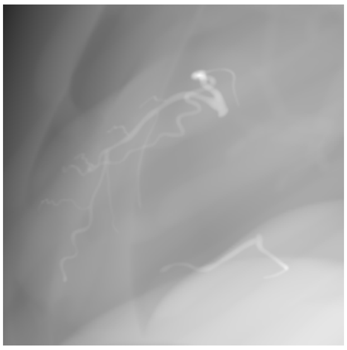
to be minimized w.r.t. the unknown segmentation Φ. The variational derivative and the gradient descent are then

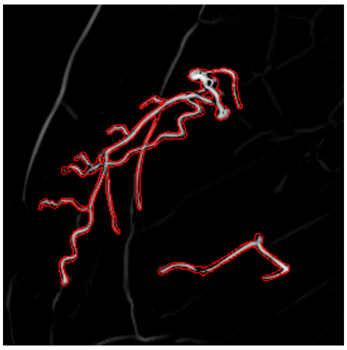
$$\delta_\Phi E\,(\Phi;\ \delta\Phi) = \int\limits_{\Omega} \delta\big(\Phi(\boldsymbol{x})\big) \cdot \Big[-\big(I(\boldsymbol{x}) - \mu_{\text{in}}\big)^2 + \big(I(\boldsymbol{x}) - \mu_{\text{out}}\big)^2\Big] \cdot \delta\Phi(\boldsymbol{x}) \ \mathrm{d}\boldsymbol{x}$$

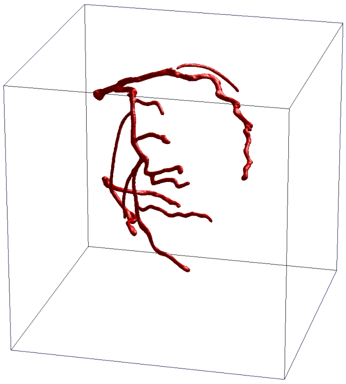
and

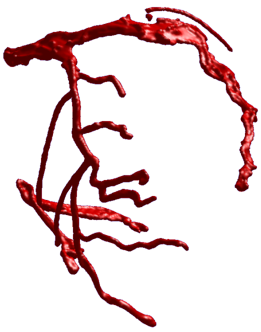
$$\frac{\partial \Phi}{\partial \tau} = \Big[\big(I(\boldsymbol{x}) - \mu_{\text{in}}\big)^2 - \big(I(\boldsymbol{x}) - \mu_{\text{out}}\big)^2\Big] \cdot \delta\big(\Phi(\boldsymbol{x})\big)$$

resp. So whenever $I(\boldsymbol{x})$ is closer to μ_{in} than μ_{out}, the latter addend will be dominating and driving the level set function $\Phi(\boldsymbol{x})$ towards lower values, i.e. to the inside. The analogous argument applies to "outside intensities" (driving the segmentation to the outside).







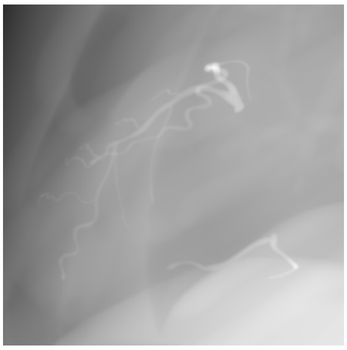


4.5 Further Reading

An overview of level set methods is best obtained by looking at textbooks like Osher and Paragios [123], Osher and Fedkiw [121], or Sethian [150]. The two websites of Sethian [146] and Fedkiw [45] contain lots of introductory information and visual results, resp., and the two author's joint paper [122] also gives an overview on level set methods, esp. for use in physical simulations. Cremers et al. [36] give a nice review on level set methods using statistical approaches and different segmentation cues.

Early developments in level set evolution techniques for modeling physical problems can be found in Osher and Sethian [124], Sethian [147], and Sethian and Straint [152].

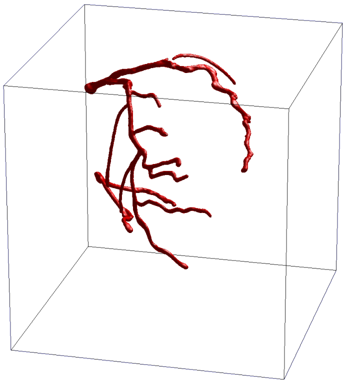
Other highly referenced work in the field is (in chronological order) Caselles et al. [20], Malladi et al. [108], Adalsteinsson and Sethian [2], Sethian [148], Malladi et al. [109], Zhao et al. [180], Caselles et al. [21], Xu and Prince [171], Adalsteinsson and Sethian [3], Peng et al. [126], Sethian [151], and Vese and Chan [166].



Chapter 5

Shape Reconstruction Using Level Sets

This chapter focuses on the presentation of the new functionals that have been developed for reconstructing an implicitly represented shape from given probabilistic projection images. In order to simplify the presentation, to follow the chain of developments, and to give first examples, we will assume a static setting throughout this chapter so that the problem effectively reduces to a multi-view reconstruction with consistent/synchronized projections (a.k.a. *shape-from-(probabilistic)-silhouette* or *shape-from-X*). Of course this problem is well-known and has been treated thoroughly in the context of level set reconstruction (as will be reviewed in Section 5.1). But keeping in mind the final application with its dynamic and non-consistent projections, the derivation of new reconstruction functionals still makes sense. Before going into details, the setting for this chapter is defined in



Problem 5.1 (Static shape reconstruction)
Let

- $\mathcal{V} \subset \mathbf{R}^3$ *be a bounded volume,*
- $\mathcal{A} \subset \mathbf{R}^2$ *be a bounded projection area, and*
- $\mathcal{T} \subset \mathbf{R}$ *be a time interval.*

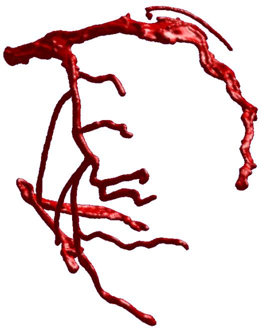
Given

- *images* $\tilde{I} \colon \mathcal{A} \times \mathcal{T} \to [0,1]$ *and*
- *projection operators* $\boldsymbol{P} \colon \mathcal{V} \times \mathcal{T} \to \mathcal{A}$,

reconstruct the implicit level set representation

$$\varPhi \colon \mathcal{V} \to \mathbf{R}$$

of a static *shape assuming that for all pixels* $\boldsymbol{x} \in \mathcal{A}$ *and for all* $t \in \mathcal{T}$ *the image value* $\tilde{I}(\boldsymbol{x}, t)$ *is related to the probability that the corresponding ray* $\boldsymbol{P}^{-1}(\boldsymbol{x}, t)$ *intersects the shape represented by* $\{\varPhi < 0\}$. *(See Figure 5.1 for an illustration.)*



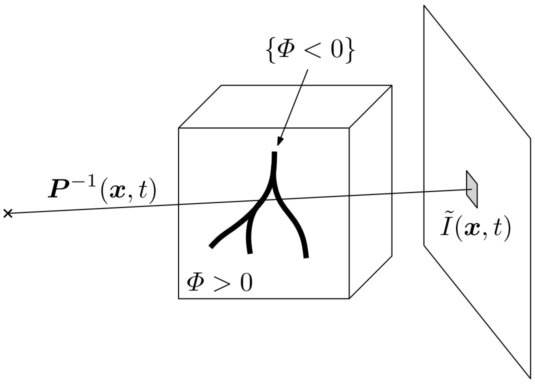

$\{\Phi < 0\}$
$\boldsymbol{P}^{-1}(\boldsymbol{x}, t)$
$\tilde{I}(\boldsymbol{x}, t)$
$\Phi > 0$

Figure 5.1: Illustration of Problem 5.1

Remark. Note that the projection operators $\boldsymbol{P}$ and images $\tilde{I}$ depend on a time variable t but the object $\{\Phi < 0\}$ to be reconstructed is *static* in Problem 5.1. (Φ does not depend on the time t.) But we prefer to use the time-dependent notation for different camera views instead of an index notation so as to stay consistent with the other parts of this work.

The values or intensities of $\tilde{I}$ could be drawn from almost any probability distribution, including the binary distribution $\{0, 1\}$. The heuristic derivation in Section 5.2 assumes two more or less symmetric probability distributions for "hitting" and "non-hitting" rays whereas the probabilistic derivation in Section 5.3 works with any given intensity distributions.

5.1 Related Work

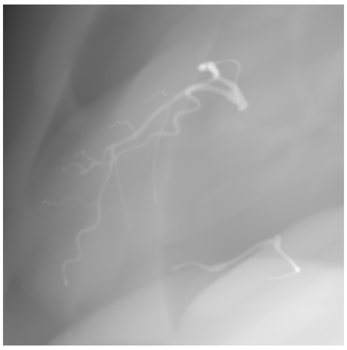
Assuming that the shape Φ is static (which it will not be in our final models and application), Problem 5.1 is of the type *shape-from-silhouette* and has been treated (also using level set methods) before. The simplest forms of reconstruction from projections employ triangulation using point correspondences or epipolar geometry (see, e. g., Hartley and Zisserman [64]) which results in a reconstructed point cloud. The remainder of this section presents an overview of several other methods for shape reconstruction.

Shape from silhouette

Shape from silhouette works on projections of the full scene instead of single points only. It can be interpreted either as a hard carving of inconsistent voxels or a thresholded unfiltered backprojection of object pixels. Baumgart's polyhedral shape models [8] and Szeliski's octree approach [164] are often cited as a historical first shape-from-silhouette algorithms. Franco and Boyer [47] present one of the latest examples on shape reconstruction. In this case, the multi-camera setting and the optical input images enable a 4D reconstruction using space occupancy grids. Cheung et al. [26, 27] are also concerned with 4D reconstruction from multiple views but restricted to articulated motions. They also use photoconsistency as an additional criterion.

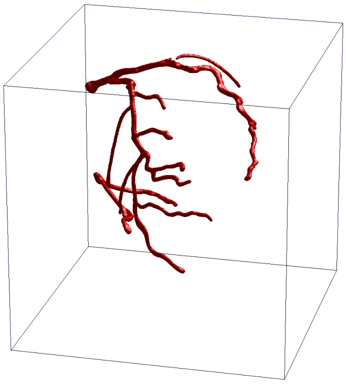
Kolev et al.

Kolev et al. recently presented various ways to formulate static shape reconstruction problems from projection data which are either of probabilistic silhouette type or consist of optical images (enabling the inclusion of photoconsistency as an optimization criterion). In the first paper in this series of works, Kolev et al. [93] use probabilistic views and a voxel-wise formulation. This is, as we will see later, not very well suited in a dynamic setting where the (initially) inconsistent images would very likely yield the trivial solution of an empty reconstruction. Kolev et al. [94] then add photoconsistency as an optimization criterion and restrict the space of level set functions to binary values using constraints. The constrained problem is then translated back to an unconstrained one using a penalizer suggested by Chan et al. [24]. Kolev and Cremers [92] only retain photoconsistency as a target functional, formulating silhouette consistency as well as the restriction to binary functions as constraints. These are enforced by intermittent projections of the optimization variables onto the set of admissible functions. Having the photoconsistency assumption as target functional allows the silhouette consistency to be moved to the constraints. Finally, Kolev et al. [95] summarize the previously mentioned work and give detailed evaluations of different energy functionals as well as a comparison with discrete optimization techniques (graph cuts). The authors give a number of arguments in favor of continuous shape models (see also Klodt et al. [90]).



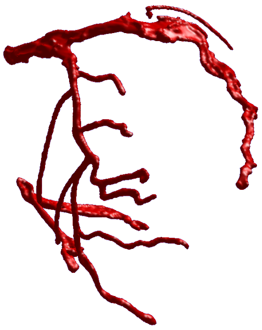
Photometric shape reconstruction

The photoconsistency assumption mentioned before is also used in various other multi-view shape reconstruction approaches. Of these, we only refer the reader to the work of two groups which also base their stereoscopic reconstructions on level set methods. Yezzi and Soatto [174, 175] and Jin et al. [78, 79] model the radiance functions for the object to be reconstructed as well as for the background. The group of Pons, Keriven, and Faugeras [44, 127, 128] base their multi-view scene reconstruction on image matching, where the last work even performs a scene flow (3D motion vector field) estimation.



Summary

Many of the methods discussed above have proven to work well in the static setting described in Problem 5.1, esp. when photometric consistency is available. Our final goal, however, is to reconstruct a dynamic scene with inconsistent projections, impeding the use of voting schemes based on the visibility of objects to rays. Trying to find a good threshold for the minimum number of votes characterizing an object or foreground point seems quite hard if the projection data is initially inconsistent (due to the unknown motion). Only later during a dynamic reconstruction would the obtained motion information bring the input projections into a consistent state. Therefore, any voting-based approach would have to take this gradual change in camera agreement into account and include some kind of heuristic for adapting the voting threshold.



Another approach is to work ray-based instead of volume-based and penalize false (positive or negative) reconstructions depending on the dissimilarity between input pixels and forward projections along the corresponding rays. We will first give a heuristic derivation of such penalty terms before presenting the more rigorous probabilistic derivation in Section 5.3.

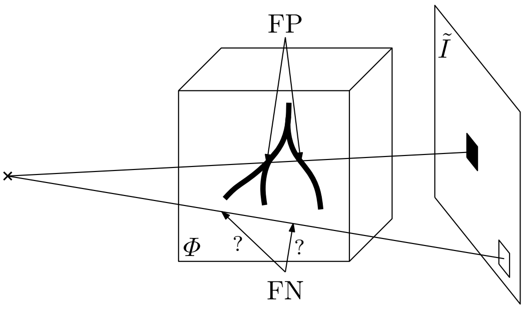


Figure 5.2: False positive and false negative reconstructions. Note that the image $\tilde{I}$ is assumed to contain high intensity or probability values for projections of vessels. Therefore a black pixel (representing low values) is a mismatch with the reconstructed (false positive) vasculature $\{\Phi < 0\}$ (represented by black voxels). Analogously, high image values and rays with positive Φ values characterize false negatives.

5.2 Heuristic Derivation

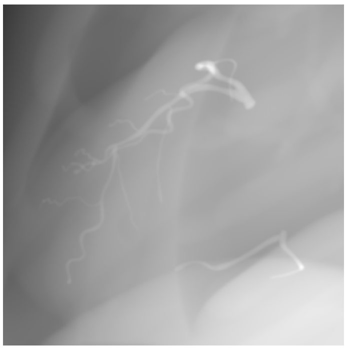
The goal of this derivation is to develop Chan-Vese-like energy functionals for the reconstruction of shapes from probabilistic silhouette data. Employing two antagonists in the energy functional just like in the intensity-based image segmentations of Chan and Vese [22, 23] yields more robust reconstructions in an initially inconsistent setting such as the dynamic scene to be treated later. The intention of including the more heuristic derivations here is to build a bridge from the Chan-Vese segmentation, which is well-known in the level set domain, to the probabilistic reconstruction functionals used in the final cardiac 4D reconstruction algorithm. The formulations given here are modeled in a more direct way and are easier to interpret than, but are also related to, the probabilistic functionals.

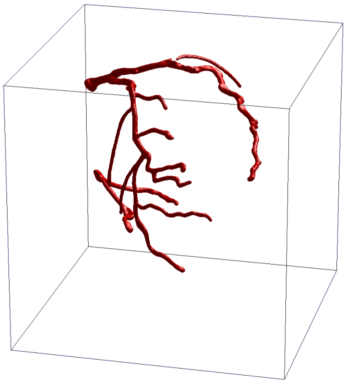
Assume we want to penalize false positive reconstructions. *False positive* in this case means that a lumen has been reconstructed ($\Phi < 0$) but a projection image's low intensity $\tilde{I}$ suggests that there should not be any lumen along the projection ray. The functional

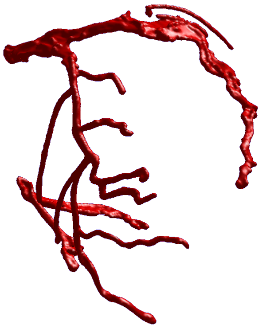
$$\int_{\mathcal{T}}\int_{\mathcal{V}} \left[1 - H\big(\Phi(\boldsymbol{X})\big)\right] \cdot H\left(\frac{1}{2} - \tilde{I}\big(\boldsymbol{P}(\boldsymbol{X},t),t\big)\right) \cdot \Big[1 - \tilde{I}\big(\boldsymbol{P}(\boldsymbol{X},t),t\big)\Big] \;\mathrm{d}\boldsymbol{X}\;\mathrm{d}t \tag{5.1}$$

achieves the desired penalization and its minimization should suppress flase positives. The first factor selects only points in the reconstruction that are located *inside* the shape of the reconstructed lumen, the second factor ensures that the whole term is only activated for points that project onto a low intensity in a projection $\tilde{I}\big(\boldsymbol{P}(\cdot,t),t\big)$, and the last factor applies a linear penalty (see Figure 5.2 for an illustration). This false positive term is a valid candidate for a reconstruction energy but trying to construct a false negative penalty in the same way does not correctly model the problem. The functional

$$\int_{\mathcal{T}}\int_{\mathcal{V}} H\big(\Phi(\boldsymbol{X})\big) \cdot H\left(\tilde{I}\big(\boldsymbol{P}(\boldsymbol{X},t),t\big) - \frac{1}{2}\right) \cdot \tilde{I}\big(\boldsymbol{P}(\boldsymbol{X},t),t\big) \;\mathrm{d}\boldsymbol{X}\;\mathrm{d}t \tag{5.2}$$







seems to be analogous to (5.1), adding a penalty for *outside* points with a high projected intensity. However, the penalties in (5.2) are activated for *all outside* points $\boldsymbol{X}$ as soon as one of their projected values has a high intensity. This would drive all points along a "high intensity ray" toward the inside of the reconstruction, although a single reconstructed lumen location along the ray would be sufficient (see also the lower ray Figure 5.2).

Therefore, instead of focusing on single volumetric points as in (5.2), one has to take *all points along a ray* into account for selecting false negative reconstructions. Observing that the minimum of multiple values is only negative if at least one of these values is negative, false negatives can be penalized by

$$\int_{\mathcal{T}}\int_{\mathcal{A}} H\left(\min_{\boldsymbol{X}\in\boldsymbol{P}^{-1}(\boldsymbol{x},t)} \Phi(\boldsymbol{X})\right) \cdot H\left(\tilde{I}(\boldsymbol{x},t) - \frac{1}{2}\right) \cdot \tilde{I}(\boldsymbol{x},t) \;\mathrm{d}\boldsymbol{x}\;\mathrm{d}t \quad . \tag{5.3}$$

This term now works pixel-based instead of voxel-based, enabling the assessment of whole rays corresponding to a pixel location $\boldsymbol{x}$. It is only activated (by the first factor) for a *ray* defined by $\boldsymbol{x}$ and t if this ray $\boldsymbol{P}^{-1}(\boldsymbol{x},t)$ does not contain any reconstructed point $\boldsymbol{X}$ with $\Phi(\boldsymbol{X}) < 0$.

Putting together (5.1) and (5.3) yields the mixed reconstruction functional

$$\begin{aligned}&\int_{\mathcal{T}}\int_{\mathcal{V}} \left[1 - H(\Phi(\boldsymbol{X}))\right] \cdot H\left(\frac{1}{2} - \tilde{I}(\boldsymbol{P}(\boldsymbol{X},t),t)\right) \cdot \left[1 - \tilde{I}(\boldsymbol{P}(\boldsymbol{X},t),t)\right] \;\mathrm{d}\boldsymbol{X}\;\mathrm{d}t \\ &\quad + \lambda \cdot \int_{\mathcal{T}}\int_{\mathcal{A}} H\left(\min_{\boldsymbol{X}\in\boldsymbol{P}^{-1}(\boldsymbol{x},t)} \Phi(\boldsymbol{X})\right) \cdot H\left(\tilde{I}(\boldsymbol{x},t) - \frac{1}{2}\right) \cdot \tilde{I}(\boldsymbol{x},t) \;\mathrm{d}\boldsymbol{x}\;\mathrm{d}t \end{aligned} \tag{5.4}$$

with a scalar weighting λ used to balance the two competing antagonists. It seems rather anomalous to have such a mixed integration (once over the volume and once over the image plane) and one might be inclined to use the ray-based treatment for both penalty components, i. e., to also integrate over the plane and use the ray $\boldsymbol{P}^{-1}(\boldsymbol{x},t)$ in the very first term in (5.4). Both options are valid with the latter one being mathematically more pleasing. The derivation and implementation of the resulting reconstruction updates, however, depend on some weighting factors (adjusting the false positive to false negative penalty as well as during the implementation of updates occurring from the minimum functional). These parameters can and should be chosen so that both approaches result in the same numeric updates.

5.3 Probabilistic Derivation

Rather than selecting and penalizing false reconstructions, one can also "think positive" and reward matching reconstructions instead. This can be achieved by maximizing the probability that the measured data and the estimated reconstruction occur together. The *maximum likelihood* and *maximum a posteriori* probabilities are two such measures and only differ in the inclusion/exclusion of prior knowledge about the probability of specific realizations of the target variables. Since we will also introduce shape priors in Chapter 7, we will follow the latter approach:

Given measurements (images) $\tilde{I}$ and a reconstruction $\varPhi$, we seek to obtain

$$
\begin{aligned}
\arg\max_{\varPhi} p(\varPhi \mid \tilde{I}) &= \arg\max_{\varPhi} \frac{p(\tilde{I} \mid \varPhi) \cdot p(\varPhi)}{p(\tilde{I})} && (5.5)\\
&= \arg\max_{\varPhi} \Big\{ p(\tilde{I} \mid \varPhi) \cdot p(\varPhi) \Big\} && (5.6)\\
&= \arg\max_{\varPhi} \left\{ p(\varPhi) \cdot \prod_{\mathcal{T}} \prod_{\mathcal{A}} p\big(\tilde{I}(\boldsymbol{x}, t) \mid \varPhi\big)^{\,\mathrm{d}\boldsymbol{x}\ \mathrm{d}t} \right\} && (5.7)\\
&= \arg\min_{\varPhi} \left\{ -\ln p(\varPhi) - \int_{\mathcal{T}}\int_{\mathcal{A}} \ln p\big(\tilde{I}(\boldsymbol{x}, t) \mid \varPhi\big)\ \mathrm{d}\boldsymbol{x}\ \mathrm{d}t \right\} && (5.8)
\end{aligned}
$$

which can be written as

$$
\arg\min_{\varPhi} \Big\{ E_{\text{prior}}(\varPhi) + E_{\text{data}}(\varPhi, \tilde{I}) \Big\}
$$

using the shorthand notations

$$
\begin{aligned}
E_{\text{prior}}(\varPhi) &:= -\ln p(\varPhi) && (5.9)\\
E_{\text{data}}(\varPhi, \tilde{I}) &:= \int_{\mathcal{T}}\int_{\mathcal{A}} -\ln p\big(\tilde{I}(\boldsymbol{x}, t) \mid \varPhi\big)\ \mathrm{d}\boldsymbol{x}\ \mathrm{d}t && (5.10)
\end{aligned}
$$

for the prior and maximum likelihood probabilities. In (5.5) we used Bayes' Theorem to interchange the random variables $\varPhi$ and $\tilde{I}$, then, noting that the denominator does not depend on the target variable $\varPhi$ leads to (5.6), and assuming i. i. d. (independence and identical distribution) leads to (5.7). Finally, the application of minus ln converts the maximization to a minimization and the continuous products (see Definition B.4) become integrals in (5.8).

Remark. Note that the i. i. d. assumption is generally not met. Though the intensities might be identically distributed, they are not independent. Neighboring image pixels are correlated since they originate from intersecting largely common anatomy. The same correlation can be present between coinciding pixels across projections. However, it is still common practice to assume independence during problem formulation.

The continuous products and integrals in (5.7) and (5.8) are evaluated over *area elements* or *pixels* (and not over volume elements as it is common practice in computer vision algorithms for multi-view reconstruction problems). Instead of counting the number of images that classify a point as object, we work projection-based and consider the ray voxels corresponding to a specific pixel. This approach avoids setting a voting threshold which would be extremely difficult for a dynamic single view setting with initially inconsistent projections.

The prior energy term E_{prior} is application-specific and will not be investigated further at this point. Thus the remaining step is to find a formulation for the conditional probabilities in the data term (5.10). Since we seek to model probability density functions (PDFs) for pixel intensities $\tilde{I}(\boldsymbol{x}, t)$, we have to investigate the corresponding ray in the reconstruction $\varPhi$ and distinguish those rays hitting the reconstructed shape from those that only traverse empty space.

Determining whether or not such a ray is hitting the reconstructed shape can be formulated using the minimum functional: The switching term

$$S(\boldsymbol{x}, t, \Phi) := H\left(\min_{\boldsymbol{X} \in \boldsymbol{P}^{-1}(\boldsymbol{x}, t)} \Phi(\boldsymbol{X})\right) \tag{5.11}$$

evaluates to 1 if *all* level set values along the ray $\boldsymbol{P}^{-1}(\boldsymbol{x}, t)$ are positive (and thus the shape was not hit) and to 0 if *any* point along the ray has a negative level set value (i. e. the shape is intersected). Defining the two sets

$$\begin{aligned} S_{\text{hit}}(\Phi) &:= \{(\boldsymbol{x}, t) \in \mathcal{A} \times \mathcal{T} \colon S(\boldsymbol{x}, t, \Phi) = 0\} \quad \text{and} \\ S_{\text{nohit}}(\Phi) &:= \{(\boldsymbol{x}, t) \in \mathcal{A} \times \mathcal{T} \colon S(\boldsymbol{x}, t, \Phi) = 1\} \end{aligned}$$

induces a partition of the image data defined on $\mathcal{A} \times \mathcal{T}$ and a conditional integration in (5.10) so that

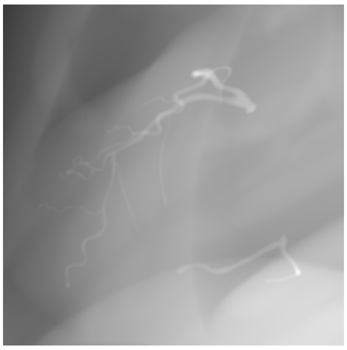
$$\boxed{\begin{aligned} E_{\text{data}}(\Phi, \tilde{I}) = & \iint\limits_{S_{\text{hit}}(\Phi)} -\ln p_{\text{hit}}\big(\tilde{I}(\boldsymbol{x}, t)\big) \;\mathrm{d}\boldsymbol{x}\;\mathrm{d}t \;+ \\ & + \iint\limits_{S_{\text{nohit}}(\Phi)} -\ln p_{\text{nohit}}\big(\tilde{I}(\boldsymbol{x}, t)\big) \;\mathrm{d}\boldsymbol{x}\;\mathrm{d}t \\ = & - \iint\limits_{\mathcal{T}\,\mathcal{A}} [1 - S(\boldsymbol{x}, t, \Phi)] \cdot \ln p_{\text{hit}}\big(\tilde{I}(\boldsymbol{x}, t)\big) \;+ \\ & + \quad S(\boldsymbol{x}, t, \Phi) \cdot \ln p_{\text{nohit}}\big(\tilde{I}(\boldsymbol{x}, t)\big) \;\mathrm{d}\boldsymbol{x}\;\mathrm{d}t \end{aligned}} \tag{5.12}$$

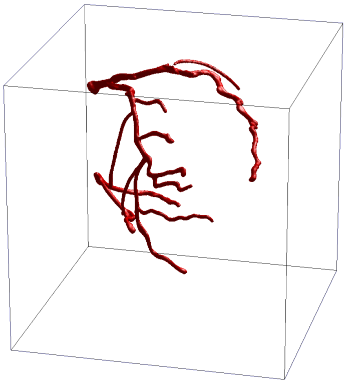
This is the final data term to be minimized, assuming there are given probability distributions p_{hit} and p_{nohit} for the image intensities of rays *hitting* and *not hitting* a lumen point, resp. Such distributions can be obtained by computing image intensity statistics for the two classes of pixels from the preprocessed angiographic projections. For the experiments performed in Chapter 8, it was sufficient to assume two half-Gaussian intensity distributions like

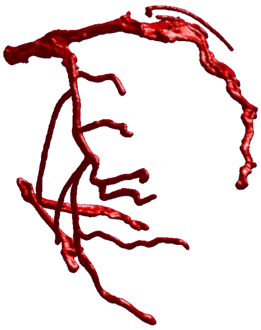
$$\begin{aligned} p_{\text{hit}} &= 2 \cdot \mathcal{N}_{1,\sigma_{\text{hit}}} \quad \text{and} \\ p_{\text{nohit}} &= 2 \cdot \mathcal{N}_{0,\sigma_{\text{nohit}}} \quad , \end{aligned}$$

with the factor 2 normalizing the two distributions to $\int_0^1 p_{(\text{no})\text{hit}} = 1$.

Remark. The first and second term on the right hand sides of (5.12) can be interpreted as false positive and false negative penalties, resp. If p_{hit} and p_{nohit} are chosen appropriately, the implementation of the probabilistic data term is actually equivalent to the heuristic one presented in the preceding section. However, the probabilistic derivation enables a better modeling of image intensities and also does not require a weighting of two competing terms. It is parameter-free, assuming p_{hit} and p_{nohit} are given.







Derivatives and updates

When calculating the variational derivative of E_{data} w. r. t. the unknown Φ, one observes that the switching term S is the only part of (5.12) depending on Φ. Using the chain rule and Corollary D.5, we first obtain

$$\delta_\Phi S\,(\boldsymbol{x}, t, \Phi;\; \delta\Phi) = \delta\left(\min_{\boldsymbol{X} \in \boldsymbol{P}^{-1}(\boldsymbol{x}, t)} \Phi(\boldsymbol{X})\right) \cdot \min_{\boldsymbol{X} \in M(\boldsymbol{x}, t, \Phi)} \delta\Phi(\boldsymbol{X})$$

with

$$M(\boldsymbol{x}, t, \Phi) = \arg \min_{\boldsymbol{X} \in \boldsymbol{P}^{-1}(\boldsymbol{x},t)} \Phi(\boldsymbol{X})$$

being the set of locations where the level set function attains its minimum along the ray $\boldsymbol{P}^{-1}(\boldsymbol{x}, t)$. This enables us to finally compute

$$\boxed{\begin{aligned} \delta_\Phi E_{\text{data}}\left(\Phi, \tilde{I};\ \delta\Phi\right) = \int\limits_{\mathcal{T}}\int\limits_{\mathcal{A}} & \left[\ln p_{\text{hit}}\big(\tilde{I}(\boldsymbol{x}, t)\big) - \ln p_{\text{nohit}}\big(\tilde{I}(\boldsymbol{x}, t)\big)\right] \cdot \\ & \cdot \delta\left(\min_{\boldsymbol{X} \in \boldsymbol{P}^{-1}(\boldsymbol{x},t)} \Phi(\boldsymbol{X})\right) \cdot \\ & \cdot \min_{\boldsymbol{X} \in M(\boldsymbol{x},t,\Phi)} \delta\Phi(\boldsymbol{X})\ \mathrm{d}\boldsymbol{x}\ \mathrm{d}t \quad . \end{aligned}} \tag{5.13}$$

Remark. A word about the notation used for variational derivatives throughout this thesis from now on: The full notation $\delta_\Phi E\,(\Phi;\ \delta\Phi)$ for a variation including the *test function* or *direction* (usually denoted by $\delta\Phi$) is used. Although it would be very practical to use the short notation $\frac{\delta E}{\delta \Phi}$, this would not allow the specification of chains of derivatives if the outer derivative cannot be expressed in $\mathcal{L}^2$ form, i. e., the outer function has Gâteaux derivatives but is not differentiable (like the data term E_{data} above). In this case, the variational derivative of the outer function can be obtained by substituting the derivative of the inner function for $\delta\Phi$.

Since the derivative (5.13) is not an inner product with $\delta\Phi$, there is no “gradient direction” and choosing a meaningful update function is not straightforward. It is clear from (5.13) that variations of $\delta\Phi$ at any location $\boldsymbol{X} \notin M(\boldsymbol{x}, t, \Phi)$ have no impact at all on the energy value E_{data}. Therefore, it makes sense to choose

$$\begin{aligned} \frac{\partial \Phi}{\partial \tau}(\boldsymbol{X}) = -\int\limits_{\mathcal{T}}\int\limits_{\mathcal{A}} & \left[\ln p_{\text{hit}}\big(\tilde{I}(\boldsymbol{x}, t)\big) - \ln p_{\text{nohit}}\big(\tilde{I}(\boldsymbol{x}, t)\big)\right] \cdot \\ & \cdot \delta\left(\min_{\boldsymbol{Y} \in \boldsymbol{P}^{-1}(\boldsymbol{x},t)} \Phi(\boldsymbol{Y})\right) \cdot \mathbf{1}_{M(\boldsymbol{x},t,\Phi)}(\boldsymbol{X})\ \mathrm{d}\boldsymbol{x}\ \mathrm{d}t \end{aligned} \tag{5.14}$$

for the update direction $\frac{\partial \Phi}{\partial \tau}$. Equation (5.14) effectively only accumulates updates on the set of minimum points $\boldsymbol{X} \in M(\boldsymbol{x}, t, \Phi)$, leaving other points unaffected.[1]

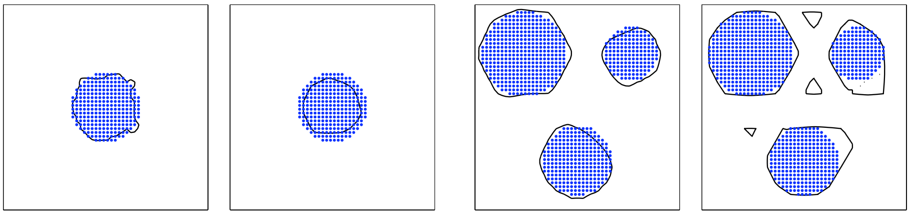
5.4 An Example: Shape from Noisy Silhouette

Figure 5.3 shows the results of our first experiments on the reconstruction of synthetic 2D shapes from noisy 1D projections. One disc (D1) and three discs (D3), resp., were reconstructed using the proposed level set reconstruction (LSR) and a standard shape-from-silhouette (SFS) algorithm. Figures 5.3(a) and (b) show that the boundary of the level set reconstruction (LSR) is not as smooth as the shape-from-silhouette (SFS) boundary but the shape fits the original object better (16 % vs. 30 % false negatives). Figures 5.3(c) and (d) demonstrate that LSR better recovers the concavities.

The quantitative errors are given in Table 5.1, where

---

[1] However, using regularization terms as well as a smooth approximation of the characteristic function $\mathbf{1}_M$ ensures that the whole level set gets updated.



(a) LSR of shape D1 (b) SFS of shape D1 (c) LSR of shape D3 (d) SFS of shape D3

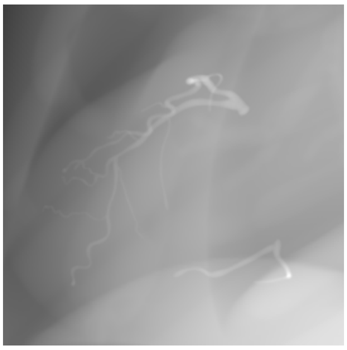
Figure 5.3: Two-dimensional shape reconstructions. Ground truth (blue) and reconstructed shape (black). (a) and (b) show the reconstructions when run on blurred and noisy data obtained from 51 projections of D1. (c) and (d) show the results obtained when run on 6 noise-free projections of D3.

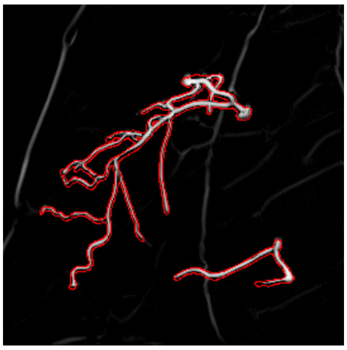
- FP is the relative number of false positives,
- FN the relative number of false negatives, and
- $\bar{d}$ the average minimum shape distance, computed using all boundary points.

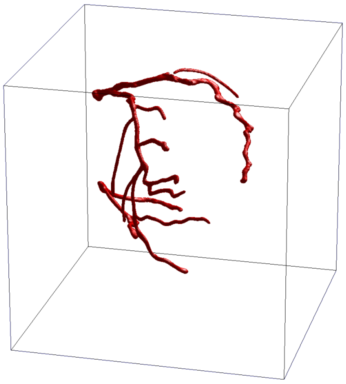
Compared to a simple shape-from-silhouette method, our level set reconstruction is more robust w. r. t. noise and limited data. It is also able to reconstruct concavities to some extent (by incorporating prior knowledge about the noise and the shape smoothness, resp.).

Figure 5.4 shows some results for 3D shape reconstruction from 2D projections.

Although our novel shape reconstruction functional is compared to a standard approach in Figure 5.3 and Table 5.1, it was not developed as a competitor to existing shape-from-silhouette methods. Depending on the application, there are certainly better ways to obtain such reconstructions. However, the given examples already show some of the versatility gained by using level sets, enabling the incorporation of all kinds of prior knowledge. But the true benefits of using implicit shape models will only surface in combination with their dynamization shown in the next chapter.







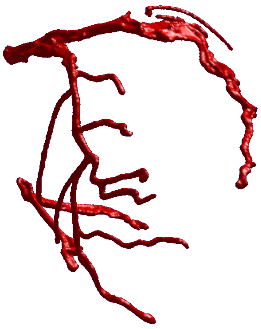
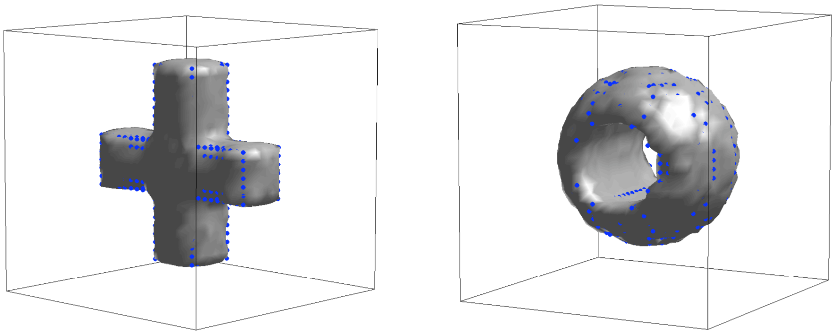


Table 5.1: Quantitative evaluation of the 2D reconstruction experiments visualized in Figure 5.3 using the proposed level set reconstruction (LSR) and standard shape-from-silhouette (SFS). False negatives (FN), false positives (FP), and average shape distance $\bar{d}$ for various combinations of ground truth shapes, number of projections, and noise levels.

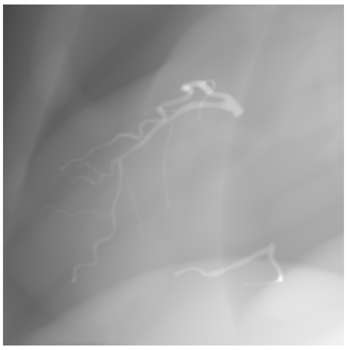
Shape	Projections	Noise	Method	FP (%)	FN (%)	$\bar{d}$ (vx)
D1	6	no	LSR	10.6	0.0	0.63
			SFS	6.3	0.0	0.40
		yes	LSR	5.5	10.9	0.68
			SFS	7.4	14.5	0.70
	51	no	LSR	0.8	2.7	0.17
			SFS	0.8	0.0	0.15
		yes	LSR	2.3	16.0	0.57
			SFS	0.0	30.1	1.41
D3	6	no	LSR	32.3	0.3	1.60
			SFS	51.1	0.0	2.41
		yes	LSR	25.1	10.6	1.33
			SFS	43.6	5.8	2.23
	51	no	LSR	7.3	1.8	0.45
			SFS	5.4	0.0	0.38
		yes	LSR	3.7	14.7	0.59
			SFS	0.1	23.4	1.03

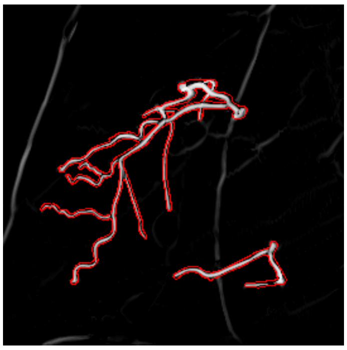


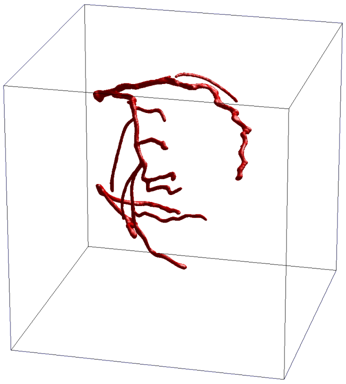
(a) Cross

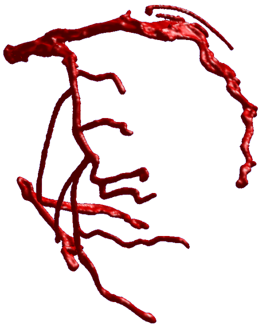
(b) Donut

Figure 5.4: Three-dimensional reconstructions. The blue dots mark the ground truth 3D shape, the gray surface the reconstructed zero level set. The projections were acquired along a circular trajectory around the objects.









Chapter 6

Dynamic Level Sets

For our final application, we need a *really dynamic*[1], volumetric level set function

$$\Phi\colon \begin{cases} \mathbf{R}^3 \times \mathcal{T} & \to \quad \mathbf{R} \\ \quad (\boldsymbol{X}, t) & \mapsto \quad \Phi\left(\boldsymbol{X}, t\right) \end{cases} \tag{6.1}$$

that represents dynamic shapes by its sign as

$$\Phi\left(\boldsymbol{X}, t\right) \begin{cases} < 0 & : \ \boldsymbol{X} \text{ is inside the shape at time } t \\ = 0 & : \ \boldsymbol{X} \text{ is at the shape's boundary at time } t \\ > 0 & : \ \boldsymbol{X} \text{ is outside the shape at time } t \quad . \end{cases}$$

We investigated two ways of modeling such dynamic level set functions by either using separate level set functions over time or by warping a reference level set function using a motion model. Both approaches have pros and cons but the latter one was chosen due to its closer resemblance to reality and ability to incorporate motion constraints. But before going into details of these approaches, we will shortly review other ways to represent dynamic shapes that are known from the literature.

6.1 Related Work

Apart from the work on motion segmentation cited in the previous chapter, the “dynamical shape priors” presented by Cremers [32, 33] should be mentioned here. In contrast to motion segmentation, this work instead builds autoregressive shape models that could be used in any tracking application where training data is available. Such data, unfortunately, is very difficult to obtain for coronary trees (see, e. g., Frangi et al. [49] for a review of techniques for the three-dimensional modeling of cardiac anatomy). Significant shape variations are quite normal as is the desire to not introduce misleading prior knowledge. This is why we will only introduce low level knowledge on the dynamic models presented in this chapter.

6.2 Fully Implicit 4D Level Set Function

The most straightforward way to obtain a dynamic level set function as in (6.1) is to assume that all level set values are independent (in space and time) and discretize or parametrize $\Phi\left(\boldsymbol{X}, t\right)$ in all four dimensions to obtain a set of level set volumes over time (see Figure 6.1). This requires a lot of memory:

---

[1] in addition to the *artificial dynamics* introduced by iterative methods

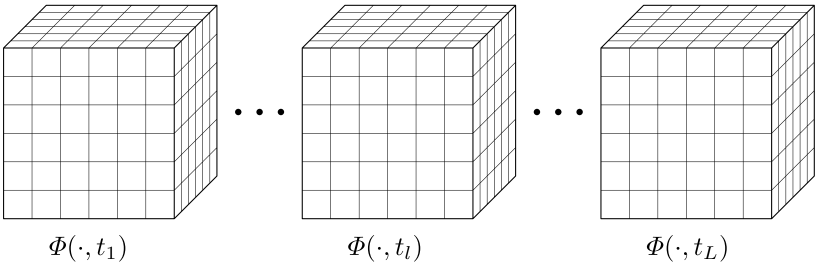


Figure 6.1: Dynamic level set as series of volumes

Assuming 200^3 voxels in space and 200 time steps this discretization requires approx. 6 GiB for each copy when stored as 4-byte floats.

Furthermore, a regularization of the shape's motion over time cannot be modeled explicitly: Any two level set values $\Phi(\boldsymbol{X}_\mathrm{a}, t_\mathrm{a})$ and $\Phi(\boldsymbol{X}_\mathrm{b}, t_\mathrm{b})$ are correspond to the same point $\boldsymbol{X}^*$ on or next to the shape if $\boldsymbol{X}_\mathrm{a} = \boldsymbol{\varphi}(\boldsymbol{X}^*, t_\mathrm{a})$ and $\boldsymbol{X}_\mathrm{b} = \boldsymbol{\varphi}(\boldsymbol{X}^*, t_\mathrm{b})$ for some motion function $\boldsymbol{\varphi}$. Identifying such correspondences among independent level set volumes, however, is generally impossible. Only the sets of surface points $\{\Phi(\cdot, t_\mathrm{a}) = 0\}$ and $\{\Phi(\cdot, t_\mathrm{b}) = 0\}$ can be identified for arbitrary times t_a and t_b (still without having a point-to-point correspondence). This would yield bulky and imperfect motion regularization terms such as

$$\int\limits_{\mathcal{T}}\int\limits_{\mathcal{T}}\int\limits_{\mathcal{V}} \frac{1}{2}\left[\delta\big(\Phi(\boldsymbol{X}, t_\mathrm{a})\big) + \delta\big(\Phi(\boldsymbol{X}, t_\mathrm{b})\big)\right] \cdot \left[\Phi(\boldsymbol{X}, t_\mathrm{a}) - \Phi(\boldsymbol{X}, t_\mathrm{b})\right]^2 \,\mathrm{d}\boldsymbol{X}\ \mathrm{d}t_\mathrm{a}\ \mathrm{d}t_\mathrm{b}$$

(which, in this case, only penalizes motions of the zero level set). Other terms for favoring periodic movements are even more complex. Also note that each of the independent shapes has to be regularized separately (so as to obtain a smooth surface etc.).

Therefore, even if a setup of independent level set volumes seems to be straightforward, simple, and versatile, an explicit model of the shape motion would be beneficial if temporal coherence is to be taken into account.

6.3 Implicit Shape and Explicit Motion

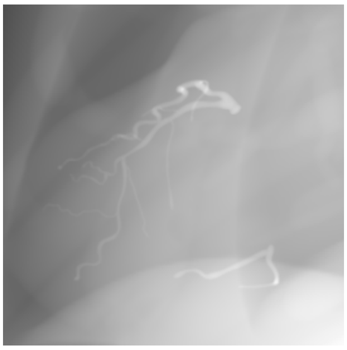
Introducing an explicit motion function enables separating shape and motion, thereby simplifying the incorporation of prior knowledge for both unknowns and possibly reducing memory requirements (if a parametrized motion model is used). Furthermore, a separate motion function enables the incorporation of a second volumetric reconstruction, e. g., a tomographic one. Such a subsequent or simultaneous reconstruction would then be coupled and consistent with the shape reconstruction (see Section 9.2).

A dynamic shape Φ is thus obtained by warping a single 3D reference level set volume Φ_0 (not necessarily linked to any instant in time) with a time-dependent motion $\boldsymbol{\varphi}$ by either postulating

$$\Phi(\boldsymbol{X}, t) = \Phi_0\big(\boldsymbol{\varphi}(\boldsymbol{X}, t)\big) \tag{6.2}$$

or

$$\Phi\big(\boldsymbol{\varphi}(\boldsymbol{X}_0, t), t\big) = \Phi_0(\boldsymbol{X}_0, t) \quad . \tag{6.3}$$



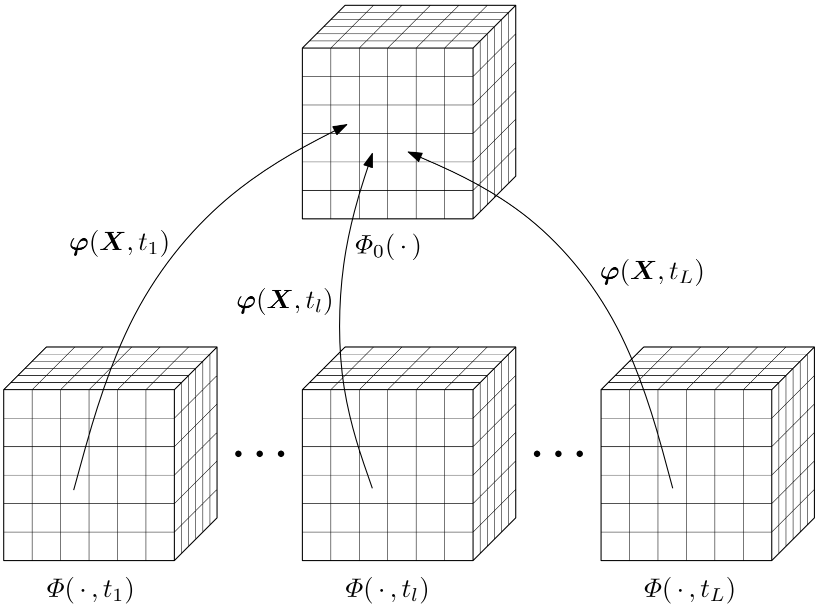
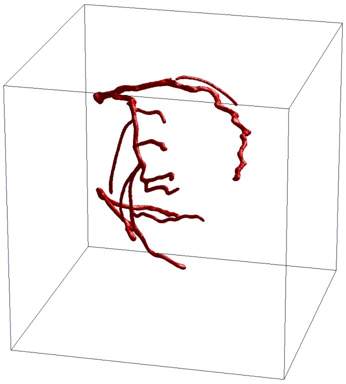

$\varphi(\boldsymbol{X}, t_1)$
$\Phi_0(\,\cdot\,)$
$\varphi(\boldsymbol{X}, t_l)$
$\varphi(\boldsymbol{X}, t_L)$
$\Phi(\,\cdot\,, t_1)$
$\Phi(\,\cdot\,, t_l)$
$\Phi(\,\cdot\,, t_L)$

Figure 6.2: Dynamic level set as warped reference volume

The former equation (see also Figure 6.2) is directly useful for interpolating the level set function Φ at time t from the reference volume Φ_0 at the computed reference point $\boldsymbol{X}_0 = \varphi\,(\boldsymbol{X}, t)$. Equation (6.3) is better-suited for interpolating in the other direction (starting from points in the reference frame and then computing the corresponding point at time t).

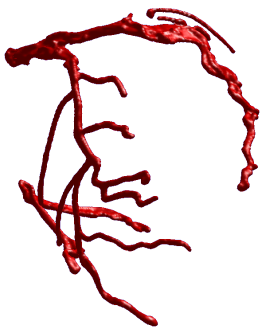
In our case, we need to work on rays through the reconstruction volume at time t. Transforming such rays into the reference frame yields a deformed line – a curve. We would therefore only be able to compute the minimum along a ray by either using (6.2) or by inverting the motion in (6.3). Since it is in general not possible to analytically invert a deformable transformation, an inversion of φ would have to be computed *iteratively for all warped volumes*. Choosing (6.2) avoids this obstacle. Therefore, we define the motion function



$$\varphi\colon \left\{ \begin{array}{ccc} \mathbf{R}^3 \times \mathcal{T} & \to & \mathbf{R}^3 \\ (\boldsymbol{X}, t) & \mapsto & \varphi(\boldsymbol{X}, t) \end{array} \right.$$

to obtain the dynamic level set function

$$\boxed{\Phi\colon \left\{ \begin{array}{ccc} \mathbf{R}^3 \times \mathcal{T} & \to & \mathbf{R} \\ (\boldsymbol{X}, t) & \mapsto & \Phi(\boldsymbol{X}, t) = \Phi_0\big(\varphi(\boldsymbol{X}, t)\big) \end{array} \right. \quad .} \tag{6.4}$$



Remark. The level set function Φ is actually only defined on the volumes

$$\mathcal{V}(t) = \varphi^{-1}(\mathcal{V}_0, t) \quad \forall\, t \in \mathcal{T}$$

due to its construction using the reference level set volume $\Phi_0\colon \mathcal{V}_0 \to \mathbf{R}$.

A last (but maybe most convincing) argument for the combination of an explicit motion with a reference volume is that a dynamic, contiguous real world

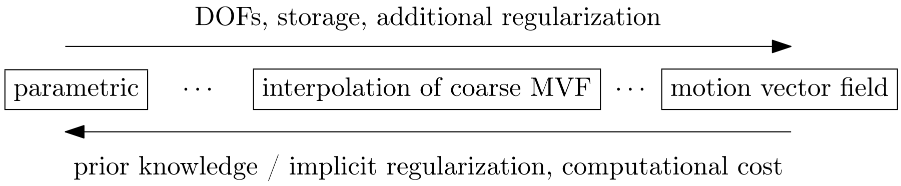

DOFs, storage, additional regularization
parametric
···
interpolation of coarse MVF
···
motion vector field
prior knowledge / implicit regularization, computational cost

Figure 6.3: Motion parametrization/discretization

object is usually much better represented by a single shape deforming over time than a set of independent objects. The "dynamization" using a warping function more closely resembles reality. This also holds true in the case of arterial tree reconstruction: Although the shape to be reconstructed seems to be sparse, one should observe that the coronaries are attached to the heart which is a contiguous structure.

The remaining question is how to model the motion $\boldsymbol{\varphi}$. The most obvious choice is a dense motion vector field (stored in a discrete grid). But one could also use a parametric motion with basis functions and a low number of degrees of freedom (DOFs). Various other motion models are possible in between these two extremes (see Figure 6.3).

6.3.1 Motion Vector Field

Storing the motion $\boldsymbol{\varphi}$ in a discrete grid of displacement vectors $\boldsymbol{\varphi}_{i,j,k,l}$ is very easy to implement since no basis functions have to be evaluated and combined for computing a voxel's displacement. And even if displacements have to be evaluated at non-grid positions, a simple linear interpolation is usually sufficient. This computationally fast implementation, however, has two significant disadvantages: The storage requirements are potentially enormous (e. g. about 18 GiB for a MVF of a 200 × 200 × 200 cube at 200 instants in time with a three-dimensional displacement vector stored in 4 B floating point numbers). Moreover, prior knowledge (about smoothness or special motion characteristics) has to be enforced using an explicit regularization which can't always be expressed by simple smoothness energy terms such as

$$\int\limits_{\mathcal{T}}\int\limits_{\mathcal{V}} \|\nabla_{\boldsymbol{X}}\boldsymbol{\varphi}\|^2 + (\partial_t\boldsymbol{\varphi})^2 \;\mathrm{d}\boldsymbol{X}\;\mathrm{d}t \quad .$$

See also [16, 14] for joint work with Blume et al. on 4D reconstruction in PET using motion vector fields.

6.3.2 Parametrized Motion

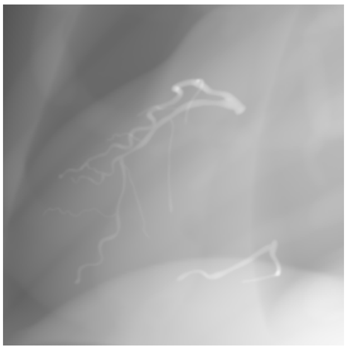
Since we have the freedom to explicitly model a motion function $\boldsymbol{\varphi}$, we can directly integrate prior knowledge for obtaining meaningful motion estimates from the optimization (thereby avoiding additional regularization terms). Similar approaches have been presented jointly with Blume et al. [15].

If the structure to be modeled is a solid object that only deforms due to external forces, for example, it makes sense to assume that rigid (or incompressible) motion components are more likely to be present than deformable ones and split the motion

$$\boldsymbol{\varphi}(\boldsymbol{X}, t) = \boldsymbol{R}(t) \cdot \boldsymbol{X} + \boldsymbol{T}(t) + \boldsymbol{u}(\boldsymbol{X}, t)$$

into a rigid (with rotation $\boldsymbol{R}$ and translation $\boldsymbol{T}$) and a deformable component $\boldsymbol{u}$ with few parameters for the latter. Since a motion is usually assumed to be smooth w. r. t. time, it also makes sense to model the temporal parameters of $\boldsymbol{R}$, $\boldsymbol{T}$, and $\boldsymbol{u}$ using spline functions. Splines allow the user to intuitively select the number of knots (using his prior knowledge) thereby reducing the "effective degrees of freedom" for the unknown variables.

As another example, one might consider special types of motions such as periodic ones. Such knowledge can also be directly integrated into an explicit motion using the appropriate basis functions.

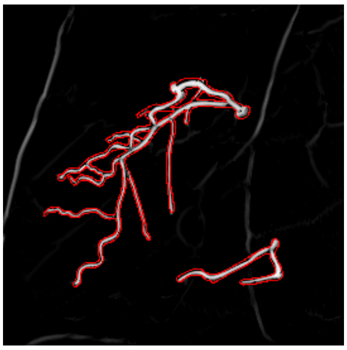


Derivatives and updates

Given a functional $E(\Phi)$ depending on a dynamic level set function $\Phi = \Phi_0 \circ \boldsymbol{\varphi}$, the final optimizations variables are Φ_0 and $\boldsymbol{\varphi}$. The derivatives of $E(\Phi_0, \boldsymbol{\varphi})$ w. r. t. these variables are obtained by using the chain rule

$$\delta_{\Phi_0} E\left(\Phi_0 \circ \boldsymbol{\varphi};\ \delta\Phi_0\right) = \delta_{\Phi} E\left(\Phi_0 \circ \boldsymbol{\varphi};\ \delta_{\Phi_0}\Phi\left(\Phi_0, \boldsymbol{\varphi};\ \delta\Phi_0\right)\right) \quad . \tag{6.5}$$

Remark. If $\frac{\delta E}{\delta \Phi}$ would be a $\mathcal{L}^2$ product in $\delta\Phi$, then the short notations for variational derivatives could be used and, e. g., (6.5) reduced to $\frac{\delta E}{\delta \Phi_0} = \left\langle \frac{\delta E}{\delta \Phi}, \frac{\delta \Phi}{\delta \Phi_0} \right\rangle$ with $\langle \cdot, \cdot \rangle$ being the inner product in $\mathcal{L}^2$.



Noting that

$$\delta_{\Phi_0}\Phi\left(\Phi_0, \boldsymbol{\varphi};\ \delta\Phi_0\right) = \delta\Phi_0 \circ \boldsymbol{\varphi} \quad ,$$

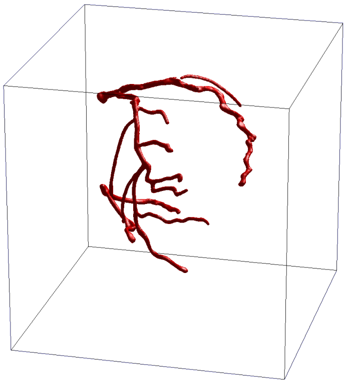
we obtain

$$\boxed{\delta_{\Phi_0} E\left(\Phi_0 \circ \boldsymbol{\varphi};\ \delta\Phi_0\right) = \delta_{\Phi} E\left(\Phi_0 \circ \boldsymbol{\varphi};\ \delta\Phi_0 \circ \boldsymbol{\varphi}\right) \quad .} \tag{6.6}$$

The other variation of interest (w. r. t. the motion $\boldsymbol{\varphi}$) can be calculated in a similar way as

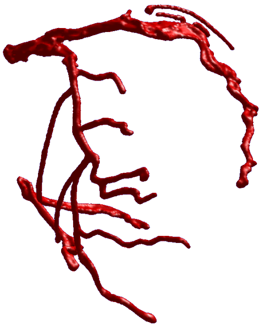
$$\boxed{\delta_{\boldsymbol{\varphi}} E\left(\Phi_0 \circ \boldsymbol{\varphi};\ \delta\boldsymbol{\varphi}\right) = \delta_{\Phi} E\left(\Phi_0 \circ \boldsymbol{\varphi};\ \langle \nabla\Phi_0 \circ \boldsymbol{\varphi}, \delta\boldsymbol{\varphi} \rangle\right) \quad .} \tag{6.7}$$

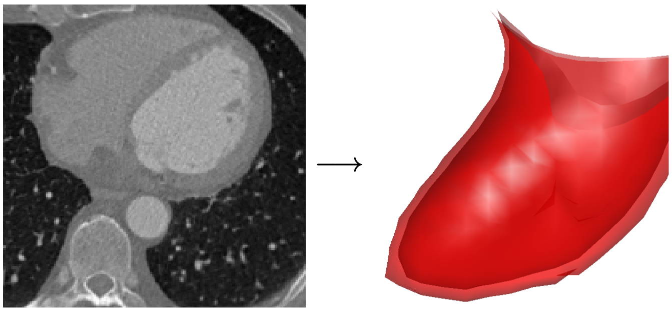
Note that warping a level set function using B-splines has been suggested before by Huang et al. [71] and Huang and Metaxas [70]. That work, however, uses the space warping induced by the splines for evolving the contour modeled by the otherwise constant level set function (used as a shape prior). Here, we modify both, the level set function as well as the warping function, for evolving the shape and the motion over artificial time.



6.4 An Application: Dynamic Segmentation

The simultaneous, intensity-based segmentation and registration of a temporal sequence of CTA volumes shown in Figure 6.4 is an example demonstrating the use of dynamic level sets in a simple application. The contrast-filled ventricles are segmented using a Chan-Vese approach [22, 23]. The only modification to this intensity-based segmentation is the dynamization of the level set function using a deformable motion in the form of a tensor product spline. So, instead of segmenting all volumes separately and trying to align them subsequently, the temporal level set enabled an integrated solution.





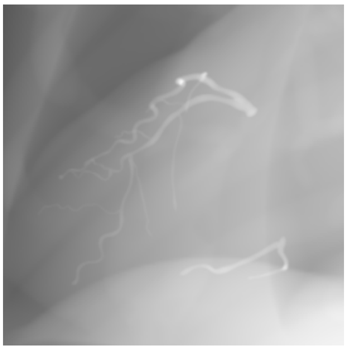
(a) Slice from the original CTA data

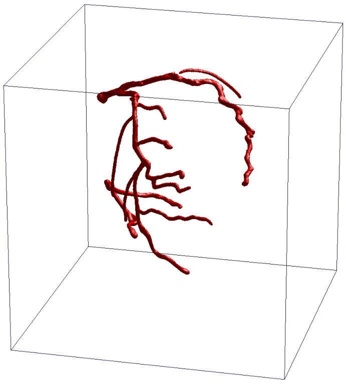
(b) Dynamic ventricle segmentation, overlay of two cardiac phases

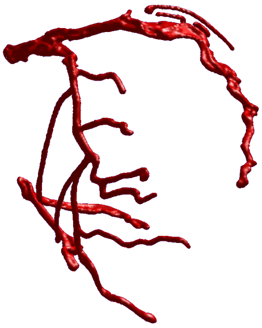
Figure 6.4: Dynamic segmentation of a 4D CTA data set

Chapter 7

Application to Cardiac Cone Beam Reconstruction







Using the dynamic level set model of the previous chapter, we are now able to reformulate Problem 5.1, seeking to reconstruct a *dynamic* shape instead of a static one:

Problem 7.1 (Dynamic shape reconstruction)
Let

- $\mathcal{V}_0 \subset \mathbf{R}^3$ *be a bounded volume,*
- $\mathcal{A} \subset \mathbf{R}^2$ *a bounded projection area, and*
- $\mathcal{T} \subset \mathbf{R}$ *be a time interval.*

Given

- *images* $\tilde{I} \colon \mathcal{A} \times \mathcal{T} \to [0, 1]$ *and*
- *projection operators* $\boldsymbol{P} \colon \mathcal{V} \times \mathcal{T} \to \mathcal{A}$,

reconstruct the reference level set

$$\Phi_0 \colon \mathcal{V}_0 \to \mathbf{R}$$

as well as the motion warping function

$$\varphi \colon \mathbf{R}^3 \times \mathcal{T} \to \mathbf{R} \quad ,$$

assuming that for all $\boldsymbol{x} \in \mathcal{A}$ *and for all* $t \in \mathcal{T}$ *the image value* $\tilde{I}(\boldsymbol{x}, t)$ *is related to the probability that the corresponding ray* $\boldsymbol{P}^{-1}(\boldsymbol{x}, t)$ *intersects the* dynamic *shape implicitly defined by*

$$\Phi = \Phi_0 \circ \varphi \quad .$$

Note that the assumptions on the input images $\tilde{I}$ require a preprocessing of the original angiographic projections I. Feature enhancement filters like those described in Frangi et al. [48], Koller et al. [96], or Krissian et al. [97] discriminate vessel and background pixels, yielding the desired relationship between $\tilde{I}$ and the "vesselness" probabilities. This dependency on a preprocessing step is an inherent limitation for any feature-based reconstruction method. The models presented in this work, however, are designed to be robust to noise or errors in preprocessed images by using continuous level set functions for matching features in space instead of a threshold-based voting algorithm.

Problem 7.1 can be characterized as follows:

- The input images are captured by a single, moving, calibrated camera imaging a dynamic scene (no multi-view setting).
- There is no photoconsistency and no optical flow since the "translucent" probability images exhibit concurrent co-located displacements.
- The object to be reconstructed undergoes non-rigid, semi-periodic[1] motions and has an unknown topology.

7.1 Combining Dynamic Models and Reconstruction Energies

Combining the shape reconstruction functional (5.12) with the dynamic level set (6.4) yields the final data term

$$\boxed{\begin{aligned} E_{\text{data}}(\varPhi_0 \circ \boldsymbol{\varphi}, \tilde{I}) = -\int_{\mathcal{T}}\!\int_{\mathcal{A}} & [1 - S(\boldsymbol{x}, t, \varPhi_0 \circ \boldsymbol{\varphi})] \cdot \ln p_{\text{hit}}\big(\tilde{I}(\boldsymbol{x}, t)\big) \; + \\ & + \; S(\boldsymbol{x}, t, \varPhi_0 \circ \boldsymbol{\varphi}) \cdot \ln p_{\text{nohit}}\big(\tilde{I}(\boldsymbol{x}, t)\big) \;\, \mathrm{d}\boldsymbol{x} \;\, \mathrm{d}t \end{aligned}} \tag{7.1}$$

with

$$S(\boldsymbol{x}, t, \varPhi) := H\left(\min_{\boldsymbol{X} \in \boldsymbol{P}^{-1}(\boldsymbol{x}, t)} \varPhi(\boldsymbol{X}, t)\right) \quad .$$

The variations of this *dynamic* shape reconstruction functional can be calculated by combining the variations derived in the previous chapters to obtain

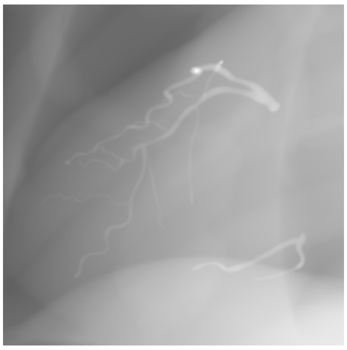
$$\boxed{\begin{aligned} \delta_{\varPhi_0} E_{\text{data}}\left(\varPhi_0 \circ \boldsymbol{\varphi}, \tilde{I};\; \delta\varPhi_0\right) & \overset{(6.6)}{=} \delta_{\varPhi} E_{\text{data}}\left(\varPhi_0 \circ \boldsymbol{\varphi}, \tilde{I};\; \delta\varPhi_0 \circ \boldsymbol{\varphi}\right) \\ & \overset{(5.13)}{=} \int_{\mathcal{T}}\!\int_{\mathcal{A}} \Big[\ln p_{\text{hit}}\big(\tilde{I}(\boldsymbol{x}, t)\big) - \ln p_{\text{nohit}}\big(\tilde{I}(\boldsymbol{x}, t)\big)\Big] \cdot \\ & \qquad \cdot \; \delta\left(\min_{\boldsymbol{X} \in \boldsymbol{P}^{-1}(\boldsymbol{x}, t)} (\varPhi_0 \circ \boldsymbol{\varphi})(\boldsymbol{X})\right) \cdot \\ & \qquad \cdot \min_{\boldsymbol{X} \in M(\boldsymbol{x}, t, \varPhi_0 \circ \boldsymbol{\varphi})} (\delta\varPhi_0 \circ \boldsymbol{\varphi})(\boldsymbol{X}) \;\, \mathrm{d}\boldsymbol{x} \;\, \mathrm{d}t \end{aligned}} \tag{7.2}$$

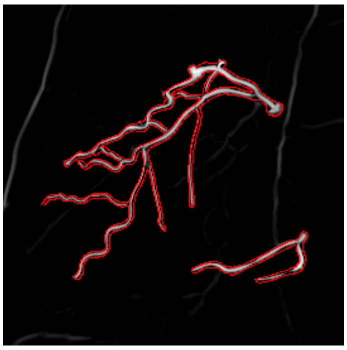
and

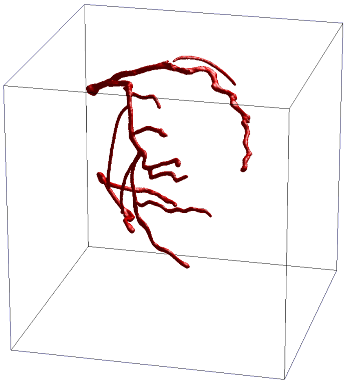
$$\boxed{\begin{aligned} \delta_{\boldsymbol{\varphi}} E_{\text{data}}\left(\varPhi_0 \circ \boldsymbol{\varphi}, \tilde{I};\; \delta\boldsymbol{\varphi}\right) & \overset{(6.7)}{=} \delta_{\varPhi} E_{\text{data}}\left(\varPhi_0 \circ \boldsymbol{\varphi}, \tilde{I};\; \langle \nabla\varPhi_0 \circ \boldsymbol{\varphi}, \delta\boldsymbol{\varphi} \rangle\right) \\ & \overset{(5.13)}{=} \int_{\mathcal{T}}\!\int_{\mathcal{A}} \Big[\ln p_{\text{hit}}\big(\tilde{I}(\boldsymbol{x}, t)\big) - \ln p_{\text{nohit}}\big(\tilde{I}(\boldsymbol{x}, t)\big)\Big] \cdot \\ & \qquad \cdot \; \delta\left(\min_{\boldsymbol{X} \in \boldsymbol{P}^{-1}(\boldsymbol{x}, t)} (\varPhi_0 \circ \boldsymbol{\varphi})(\boldsymbol{X})\right) \cdot \\ & \qquad \cdot \min_{\boldsymbol{X} \in M(\boldsymbol{x}, t, \varPhi_0 \circ \boldsymbol{\varphi})} (\langle \nabla\varPhi_0 \circ \boldsymbol{\varphi}, \delta\boldsymbol{\varphi} \rangle)(\boldsymbol{X}) \;\, \mathrm{d}\boldsymbol{x} \;\, \mathrm{d}t \quad , \end{aligned}} \tag{7.3}$$

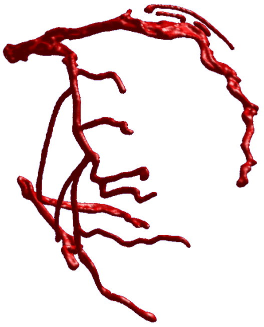
---

[1] We will use the expression *semi-periodic* to express the fact that the cardiac motion is only approximately periodic (see Achenbach [1]) and also superimposed by a residual breathing motion (that is usually of lower wavelength than the acquisition time and therefore assumed to be non-periodic.)









with

$$M(\boldsymbol{x}, t, \Phi \circ \boldsymbol{\varphi}) = \arg \min_{\boldsymbol{X} \in \boldsymbol{P}^{-1}(\boldsymbol{x},t)} (\Phi \circ \boldsymbol{\varphi})(\boldsymbol{X}, t)$$

being the set of minimum points along the ray $\boldsymbol{P}^{-1}(\boldsymbol{x}, t)$ in the warped level set volume $\Phi_0 \circ \boldsymbol{\varphi}$.

These variations are independent of the parametrization or discretization chosen in the following sections. However, it is usually not possible to apply updates at the exact set of minimum points $M(\boldsymbol{x}, t, \Phi)$ as suggested in (5.14) due to the discretization. This is why we have chosen to gradually weight the updates for a point $\boldsymbol{X}$ with its distance to M: Assuming that Φ_0 closely approximates a distance function, the term

$$d_{M(\boldsymbol{x},t,\Phi_0 \circ \boldsymbol{\varphi})}(\boldsymbol{X}) := \big|(\Phi_0 \circ \boldsymbol{\varphi})(\boldsymbol{X}) - (\Phi_0 \circ \boldsymbol{\varphi})(M)\big|$$

measures the distance of the point $\boldsymbol{X}$ to the set M. Acknowledging that discretization effects prevent the application of updates at the points in the analytic set M, it is a reasonable choice to weight updates for any point $\boldsymbol{X}$ with its distance to this set. Any inverse function on the distance d could be used. Here, we opted to "recycle" the mollified[2] Dirac functions so that the update equations have the characteristic function $\mathbf{1}_{M(\boldsymbol{x},t,\Phi_0 \circ \boldsymbol{\varphi})}(\boldsymbol{X})$ (resulting from the last factors in (7.2) and (7.3), resp.) replaced by the factor $\delta\big(d_{M(\boldsymbol{x},t,\Phi_0 \circ \boldsymbol{\varphi})}(\boldsymbol{X})\big)$.

7.2 Parametrization, Discretization, and Regularization

The very last step in feeding the reconstruction problem into a computer requires a discretization of the associated variables. There are several ways to do this and the choice directly affects the degrees of freedom (DOF) for these variables. We therefore integrate the discussion on discretization with parametrization and regularization questions.

From a practical point of view, a regularization or low-DOF parametrization smooths the results by introducing neighborhood relations. From a mathematical point of view, ill-posed problems can be turned into well-posed ones. However, regularization or restrictive modeling always entails a compromise between noise suppression and bias toward the priors and therefore has to be chosen carefully, esp. in medical applications.

7.2.1 Input Data

The input data consists of the preprocessed (i. e., vessel-enhanced) image data $\tilde{I}$, the corresponding projection operators $\boldsymbol{P}$, and the relative cardiac phase c (if an ECG signal is available), all given for a discrete set of acquisition time points t_l. Each of the projection images $\tilde{I}(\cdot, t_l)$ is also defined on a spatial grid $\boldsymbol{x}_{r,s}$ with integer pixel indexes r and s so that the input images are naturally discretized as

$$\tilde{I}(\boldsymbol{x}_{r,s}, t_l)\ , \quad r, s, l \in \mathbf{N}_0 \quad .$$

Any values of $\tilde{I}$ at non-grid positions will be interpolated.

---

[2] Yes, this word is indeed spelled "mollified" and refers to the tamed characteristics compared to the Dirac functional.

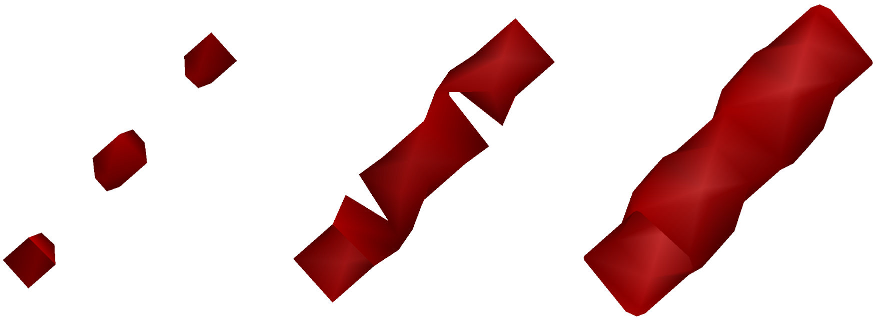


Figure 7.1: Level set resolution. Oblique tubes with diameters of 0.5 mm, 0.75 mm, and 1 mm represented using a level set grid with 0.5 mm spacing.

7.2.2 Reference Level Set – Shape

The dynamic level set Φ does not have to be fully kept in the memory. Its values are rather computed on-the-fly by mapping a requested location $\boldsymbol{X}$ at time t to $\boldsymbol{X}_0 = \varphi(\boldsymbol{X}, t)$ and interpolating the appropriate value from the reference level set Φ_0 instead.[3]

The reference level set Φ_0, in turn, can be parametrized and stored in many different ways (also depending on the amount and type of prior knowledge available). Although there are other approaches, (e. g., using radial basis functions as Morse et al. [110], Slabaugh et al. [159] or unstructured sampling points as Ho et al. [67]), the most obvious, simple, and widespread choice we also made is to discretize the level set on a rectangular grid.

The voxel size or element spacing has to be selected with the application in mind. In our case, the coronary vasculature can have diameters as small as 1 mm (see Table 1.1) and we set Φ_0's spacing to 0.5 mm. This allows to represent tubular structures with diameters in the desired range. See Figure 7.1 for an illustration of the resolution limit when implicitly representing tubular structures in 3D.

When iterating over volumetric coordinates, we will use the notation $\boldsymbol{X}_{i,j,k}$ and $\boldsymbol{X}_{0,i,j,k}$, resp., to address the point corresponding to index i, j, k.

Shape smoothness

We will not discuss detailed shape priors as there is no such prior knowledge available. The coarse notion of requesting a *smooth shape*, however, can be used to regularize the densely stored level set. A shape prior commonly chosen so as to obtain such a smooth surface results from the prior

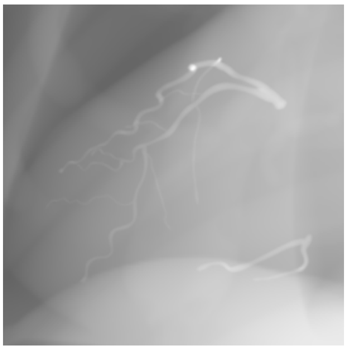
$$p_{\text{shape}}(\Phi_0) = \lambda_{\text{shape}} \cdot \mathrm{e}^{-\lambda_{\text{shape}} S(\Phi_0)} \tag{7.4}$$

where $0 \leq S(\Phi_0) < \infty$ is the shape area that can be expressed as

$$S(\Phi_0) = \left|\{\Phi_0 = 0\}\right| = \int\limits_{\{\Phi_0=0\}} 1 \ \mathrm{d}\boldsymbol{\xi} \overset{\text{Theorem E.2}}{=} \int\limits_{\mathcal{V}_0} \delta(\Phi_0) \|\nabla \Phi_0\| \ \mathrm{d}\mathcal{V}_0 \quad .$$

---

[3] Note that an algorithm may store precomputed warped level sets $\Phi(\cdot, \boldsymbol{X})$ in order to speed up the computations internally.



The exponential in (7.4) was used as an inverting function for modeling smaller shape areas (and therefore smoother shapes) as more likely. The factor $\lambda_{\text{shape}} > 0$ as introduced here does not alter p_{shape}'s classification as a probability function (with $p_{\text{shape}} \geq 0$ and $\int_0^\infty p_{\text{shape}}(S)\ \mathrm{d}S = 1$) but acts as a weighting factor in the final energy term

$$\boxed{E_{\text{shape}}(\varPhi_0) = -\ln p_{\text{shape}}(\varPhi_0) = \underbrace{-\ln\lambda_{\text{shape}}}_{\text{const.}} + \lambda_{\text{shape}} \int\limits_{\mathcal{V}_0} \delta(\varPhi_0)\|\nabla\varPhi_0\|\ \mathrm{d}\mathcal{V}_0 \quad .}$$

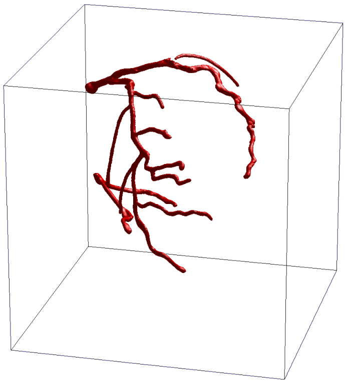
From Appendix E.4, we get

$$\boxed{\delta_{\varPhi_0} E_{\text{shape}}\left(\varPhi_0;\ \delta\varPhi_0\right) = -\lambda_{\text{shape}} \int\limits_{\mathcal{V}_0} \delta(\varPhi_0)\,\mathrm{div}\left(\frac{\nabla\varPhi_0}{\|\nabla\varPhi_0\|}\right)\delta\varPhi_0\ \mathrm{d}\boldsymbol{X}_0}$$

for the variation (see Appendix E.2 for a full equation for $\mathrm{div}\left(\frac{\nabla\varPhi_0}{\|\nabla\varPhi_0\|}\right)$) and

$$\boxed{\frac{\partial\varPhi_0}{\partial\tau} = \lambda_{\text{shape}} \cdot \delta(\varPhi_0)\,\mathrm{div}\left(\frac{\nabla\varPhi_0}{\|\nabla\varPhi_0\|}\right)}$$

for the update.

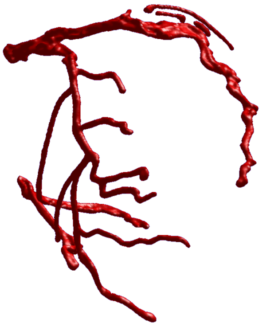


Signed distance

Even though we were not assuming that $\varPhi_0$ is a signed distance function at any point during the derivations so far,[4] it still makes sense to keep the level set function at least close to such a signed distance function. The reason is that $\varPhi_0$ develops very steep gradients (esp. when using a region-based data term that translates into a δ factor in the update equations). This introduces noisy features affecting the finite differences used when approximating spatial derivatives.

Instead of just smoothing the level set function (thereby inadvertently modifying the zero contour), one seeks to preserve the signed distance property with the zero contour remaining intact and therefore leaving the implicitly modeled shape unaffected.

There are various ways to enforce the signed distance property. The most obvious one is to extract the zero contour after every iteration (e. g., using the marching cubes of Lorensen and Cline [104] or the newer method of Kobbelt et al. [91]) and then recompute the distance values to this boundary (which can be done quite fast with the fast marching method by Sethian [149]).



Since it is usually not necessary to have the distance values exactly restored, the more common method is to iteratively reinitialize the level set function once in a while using Sussman et al.'s method [163, 162]. Solving the PDE

$$\frac{\partial\varPhi_0}{\partial\tau} = S\big(\varPhi_0(\cdot,0)\big) \cdot \left(1 - \|\nabla\varPhi_0\|\right) \tag{7.5}$$

---

[4] In fact, we will even initialize $\varPhi_0 = 0$ which is quite "far" from a distance function.

to steady state would exactly recover the signed distance function. The term

$$S\big(\Phi_0(\cdot,0)\big) = \frac{\Phi_0(\cdot,0)}{\sqrt{\Phi_0(\cdot,0)^2 + C}}$$

is, for numerical reasons, an approximation to the sign function, with C being on the order of the spatial resolution. This sign function is computed only once from the initial level set $\Phi_0(\cdot,0)$ in order to preserve the initial boundary. The PDE then drives the level set function towards a signed distance function, leaving the zero level set unaltered. Note that it is usually not necessary to compute (7.5) to steady state. Very few iterations of the reinitialization PDE are usually enough to obtain the desired regularization. This reinitialization method is also explained in detail by Zhao et al. [180] and used in a modified version by Peng et al. [126]. Both methods described so far are also covered in more detail by Osher and Fedkiw [121, Ch. 7].

A third possibility is to penalize deviations of the level set function from a signed distance function. Li et al. [101, 102] add

$$E_{\text{dist}}(\Phi_0) = \lambda_{\text{dist}} \int\limits_{\mathcal{V}_0} \frac{1}{2}(\|\nabla\Phi_0\| - 1)^2 \;\mathrm{d}\boldsymbol{X}_0 \tag{7.6}$$

to the energy functional for penalizing deviations of $\|\nabla\Phi_0\|$ from 1. The unit gradient magnitude is a necessary property of signed distance functions (though not sufficient as often suggested). This method, however, entails other numerical difficulties as it tends to make the evolution unstable. See Appendix E.5 for the calculation of the variation of (7.6) that can be used as additional update component in a gradient descent.

Other approaches for keeping the level set function a signed distance function have been presented by Gomes and Faugeras [54, 55] and Adalsteinsson and Sethian [3]. In both cases, the authors propose to directly modify the evolution equation so that the signed distance property is preserved by the updates.

7.2.3 Warping Function – Motion

If the structure to be modeled is a continuous object that deforms due to forces (e. g., originating from breathing and heart beat), it makes sense to assume that rigid (or incompressible) motion components are more likely to be present than deformable ones. This can be favored by composing the motion by a rotation $\boldsymbol{R}$, a translation $\boldsymbol{T}$, and a deformation $\boldsymbol{u}$ to obtain

$$\boldsymbol{R}(t)\cdot\boldsymbol{X} + \boldsymbol{T}(t) + \boldsymbol{u}(\boldsymbol{X},t) \quad .$$

Enforcing a smooth motion over time can be achieved by using B-splines for all the time-dependent components $\boldsymbol{R}$, $\boldsymbol{T}$, and $\boldsymbol{u}$. The rotation

$$\boldsymbol{R}(t,\boldsymbol{\alpha}) = \boldsymbol{R}_1\big(\theta_1(t)\big)\cdot\boldsymbol{R}_2\big(\theta_2(t)\big)\cdot\boldsymbol{R}_3\big(\theta_3(t)\big)$$

is parametrized by Euler angle matrices $\boldsymbol{R}_1$ to $\boldsymbol{R}_3$ whose angular parameters

$$\theta_i(t) = \sum_{b_t} \alpha_{\boldsymbol{R}_i,b_t}\cdot N^t_{b_t}(t) \quad , 1 \le i \le 3$$

are temporal B-splines with coefficient vectors $\boldsymbol{\alpha}_{\boldsymbol{R}_1}$ to $\boldsymbol{\alpha}_{\boldsymbol{R}_3}$. The translation

$$\boldsymbol{T}(t,\boldsymbol{\alpha}) = \sum_{b_t} \boldsymbol{\alpha}_{\boldsymbol{T},b_t}\cdot N^t_{b_t}(t)$$

Algorithm 7.1 Efficient evaluation of motion splines

```
1  evaluate $N^X_{b_X}(X_i)$, $N^Y_{b_Y}(Y_j)$, $N^Z_{b_Z}(Z_k)$, $N^t_{b_t}(t_l)$ at all grid points
2  for $l$ do  // projection images / time
3      $\boldsymbol{u}_{b_X,b_Y,b_Z} \leftarrow \sum_{b_t} \boldsymbol{\alpha}_{\boldsymbol{u},b_X,b_Y,b_Z,b_t} \cdot N^t_{b_t}(t_l)$
4      $\boldsymbol{R} \leftarrow \ldots$
5      $\boldsymbol{T} \leftarrow \ldots$
6      for $k$ do
7          $\boldsymbol{u}_{b_X,b_Y} \leftarrow \sum_{b_Z} \boldsymbol{u}_{b_X,b_Y,b_Z} \cdot N^Z_{b_Z}(Z_k)$
8          for $j$ do
9              $\boldsymbol{u}_{b_X} \leftarrow \sum_{b_Y} \boldsymbol{u}_{b_X,b_Y} \cdot N^Y_{b_Y}(Y_j)$
10             for $i$ do
11                 $\boldsymbol{u} \leftarrow \sum_{b_X} \boldsymbol{u}_{b_X} \cdot N^X_{b_X}(X_i)$
12                 // compute warped level set and updates to $\Phi_0$ using
13                 $\boldsymbol{X}_0 \leftarrow \boldsymbol{R} \cdot \boldsymbol{X}_{i,j,k} + \boldsymbol{T} + \boldsymbol{u}$
14             end for
15         end for
16     end for
17 end for
```

is simply a three-dimensional B-spline function with a coefficient matrix $\boldsymbol{\alpha}_{\boldsymbol{T}}$. The deformable motion $\boldsymbol{u}$ should additionally be smooth with respect to the spatial coordinates. This is achieved by not only using B-splines in the temporal dimension of $\boldsymbol{u}(\boldsymbol{X}, t)$ but also in the three spatial ones, resulting in the tensor product spline

$$\boldsymbol{u}(\boldsymbol{X}, t) = \sum_{b_X} \sum_{b_Y} \sum_{b_Z} \sum_{b_t} \boldsymbol{\alpha}_{\boldsymbol{u},b_X,b_Y,b_Z,l} \cdot N^X_{b_X}(X) \cdot N^Y_{b_Y}(Y) \cdot N^Z_{b_Z}(Z) \cdot N^t_{b_t}(t) \quad .$$

Note that all the B-spline basis functions can be pre-computed once in an algorithm if the evaluation points are always the same grid points X_i, Y_j, Z_k, and t_l resp.(see line 1 in Algorithm 7.1). Furthermore, if the summation order corresponds well with the nested loops in an algorithm, many intermediate products can be efficiently evaluated as soon as another space or time variable is known, e. g. in lines 3, 7, 9, and 11 in Algorithm 7.1. In this case an efficient order of operations would be

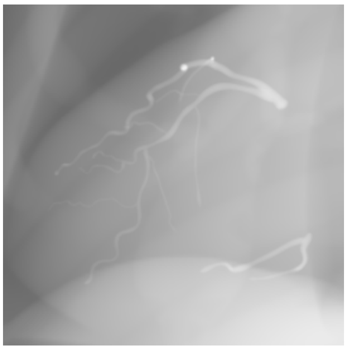
$$\boldsymbol{u}(\boldsymbol{X}_{i,j,k}, t_l) = \\ \sum_{b_X} \left(\sum_{b_Y} \left(\sum_{b_Z} \left(\sum_{b_t} \boldsymbol{\alpha}_{\boldsymbol{u},b_X,b_Y,b_Z,b_t} \cdot N^t_{b_t}(t_l) \right) \cdot N^Z_{b_Z}(Z_k) \right) \cdot N^Y_{b_Y}(Y_j) \right) \cdot N^X_{b_X}(X_i)$$

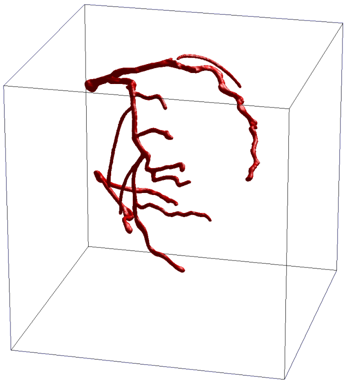
where, e. g., the term in the innermost parentheses does not depend on X, Y, or Z and can therefore be precomputed right after the outermost loop (line 2 of Algorithm 7.1) in line 3.

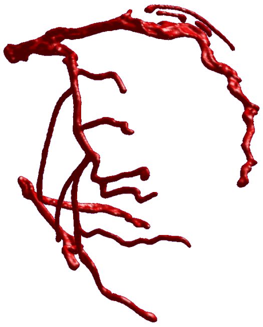
Now that the detailed model is specified, we will gather all motion parameters $\alpha_{\boldsymbol{R}_i,l}$, $\boldsymbol{\alpha}_{\boldsymbol{T},l}$, and $\boldsymbol{\alpha}_{\boldsymbol{u},i,j,k,l}$ and denote their aggregate vector by $\boldsymbol{\alpha}_{\text{glo}}$ to obtain

$$\boldsymbol{\varphi}_{\text{glo}}\big(\boldsymbol{X}, t, \boldsymbol{\alpha}_{\text{glo}}\big) = \boldsymbol{R}\big(t, \boldsymbol{\alpha}_{\text{glo}}\big) \cdot \boldsymbol{X} + \boldsymbol{T}\big(t, \boldsymbol{\alpha}_{\text{glo}}\big) + \boldsymbol{u}\big(\boldsymbol{X}, t, \boldsymbol{\alpha}_{\text{glo}}\big) \quad . \tag{7.7}$$

In addition to this time-dependent *global* motion, we also define a *periodic* motion depending on the (ECG-determined) cardiac phase $c(t) \in [0\,\%, 100\,\%)$







as

$$\boldsymbol{\varphi}_{\mathrm{per}}\big(\boldsymbol{X}, c(t), \boldsymbol{\alpha}_{\mathrm{per}}\big) = \boldsymbol{R}\big(c(t), \boldsymbol{\alpha}_{\mathrm{per}}\big) \cdot \boldsymbol{X} + \boldsymbol{T}\big(c(t), \boldsymbol{\alpha}_{\mathrm{per}}\big) + \boldsymbol{u}\big(\boldsymbol{X}, c(t), \boldsymbol{\alpha}_{\mathrm{per}}\big) \quad . \tag{7.8}$$

The cardiac motion model's periodicity is enforced by appropriately designing the temporal B-spline basis. Note that although $\boldsymbol{\varphi}_{\mathrm{per}}$ is thus exactly periodic, the combination with the global motion $\boldsymbol{\varphi}_{\mathrm{glo}}$ enables the representation of a non-ideally periodic heart beat (in addition to allowing breathing motion also captured by $\boldsymbol{\varphi}_{\mathrm{glo}}$).

There are various ways to combine the two motion models (7.7) and (7.8), namely by concatenation ($\boldsymbol{\varphi}_{\mathrm{glo}} \circ \boldsymbol{\varphi}_{\mathrm{per}}$ or $\boldsymbol{\varphi}_{\mathrm{per}} \circ \boldsymbol{\varphi}_{\mathrm{glo}}$) or by addition. We have chosen the latter approach due to its symmetry and easier implementation, arriving at

$$\boxed{\begin{aligned} &\boldsymbol{\varphi}\big(\boldsymbol{X}, t, \boldsymbol{\alpha}_{\mathrm{glo}}, \boldsymbol{\alpha}_{\mathrm{per}}\big) = \\ &= \boldsymbol{X} + \big(\boldsymbol{R}(t, \boldsymbol{\alpha}_{\mathrm{glo}}) - \boldsymbol{I}_3\big) \cdot \boldsymbol{X} + \boldsymbol{T}(t, \boldsymbol{\alpha}_{\mathrm{glo}}) + \boldsymbol{u}(\boldsymbol{X}, t, \boldsymbol{\alpha}_{\mathrm{glo}}) + \\ &\qquad + \big(\boldsymbol{R}\big(c(t), \boldsymbol{\alpha}_{\mathrm{per}}\big) - \boldsymbol{I}_3\big) \cdot \boldsymbol{X} + \boldsymbol{T}\big(c(t), \boldsymbol{\alpha}_{\mathrm{per}}\big) + \boldsymbol{u}\big(\boldsymbol{X}, c(t), \boldsymbol{\alpha}_{\mathrm{per}}\big) \quad . \end{aligned}}$$

Thus, $\boldsymbol{\alpha}_{\mathrm{glo}}$ and $\boldsymbol{\alpha}_{\mathrm{per}}$ are multidimensional variables to be optimized and are grouped in $\boldsymbol{\alpha}$. Since they are motion parameters, we can calculate gradients for them setting $\delta\boldsymbol{\varphi} = \frac{\partial \boldsymbol{\varphi}}{\partial \alpha_i}$ in the motion variation (7.3) to obtain

$$\boxed{\begin{aligned} \frac{\partial E_{\mathrm{data}}}{\partial \alpha_i} &= \delta_{\boldsymbol{\varphi}} E_{\mathrm{data}} \left(\mathit{\Phi}_0 \circ \boldsymbol{\varphi}, \tilde{I};\ \frac{\partial \boldsymbol{\varphi}}{\partial \alpha_i} \right) \\ &= \delta_{\mathit{\Phi}} E_{\mathrm{data}} \left(\mathit{\Phi}_0 \circ \boldsymbol{\varphi}, \tilde{I};\ \left\langle \nabla \mathit{\Phi}_0 \circ \boldsymbol{\varphi}, \frac{\partial \boldsymbol{\varphi}}{\partial \alpha_i} \right\rangle \right) \quad . \end{aligned}}$$

7.3 Algorithm

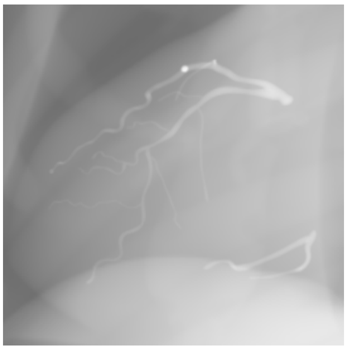
Having defined discrete representations of all the input and output variables of the continuous problem formulation we will now outline the core algorithm for computing a dynamic reconstruction using the iterative method developed in the previous chapters (see Algorithm 7.2). An implementation of the full algorithm requires additional subroutines, e. g. for computing warped volumes, casting rays, etc., which will not be discussed in detail here.

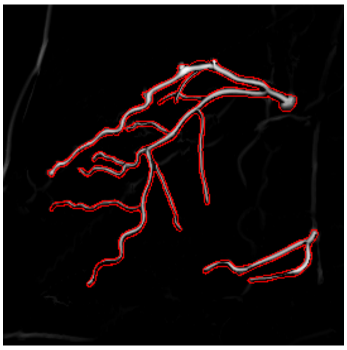
Initialization

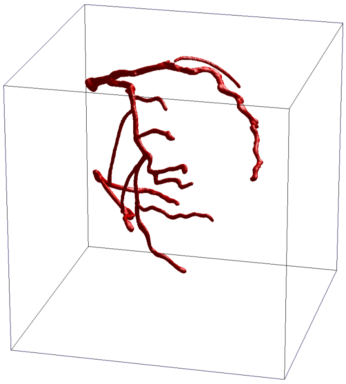
The reference level set volume is initialized to zero everywhere (see line 1 in Algorithm 7.2). This means that *no prior shape* is assumed and that the reconstruction is “undecided” at every point in space. Other publications suggest initializing the level set function so that it describes some meaningful shapes (such as a number of balls covering the entire volume) in order to start with a signed distance function. Starting with a constant zero function, however, enables a fast initial convergence since *all points* are considered to be *on the boundary* between “inside” and “outside” and updates depending on $\delta(\mathit{\Phi})$ will be effective throughout the volume. Similarly, the motion is

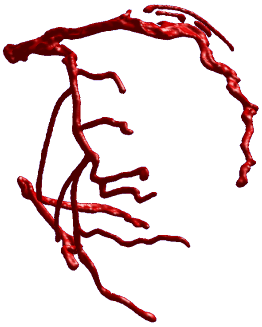
Algorithm 7.2 Dynamic cardiac reconstruction

```
1  initialize $\Phi_0 \leftarrow 0$
2  initialize $\boldsymbol{\alpha} \leftarrow 0$
3  repeat  // iterations
4      reset updates $\Delta\Phi_0 \leftarrow 0$, $\Delta\boldsymbol{\alpha} \leftarrow 0$
5      accumulate update $\Delta\Phi_0 \leftarrow \Delta\Phi_0 - \frac{\delta E_{\text{shape}}}{\delta \Phi_0}$ from shape prior
6      for $l$ do  // projection images
7          compute warped volume $\Phi(\cdot, t_l) \leftarrow \Phi_0\big(\boldsymbol{\varphi}(\cdot, t_l)\big)$
8          compute mIPs $\Phi_{\min}(\boldsymbol{x}_{r,s}, t_l) \leftarrow \min_{\boldsymbol{X} \in \boldsymbol{P}^{-1}(\boldsymbol{x}_{r,s}, t_l)} \big(\Phi(\cdot, t_l)\big) \quad \forall\, r, s$
               by casting rays $\boldsymbol{P}^{-1}(\boldsymbol{x}_{r,s}, t_l)$ through $\Phi(\cdot, t_l)$
9          for $k$ do  // $Z$ index of warped volume
10             for $j$ do  // $Y$ index of warped volume
11                 for $i$ do  // $X$ index of warped volume
12                     compute projection location $\boldsymbol{x} = \boldsymbol{P}(\boldsymbol{X}_{i,j,k}, t_l)$
13                     interpolate vesselness value $\tilde{I}^* \leftarrow \tilde{I}(\boldsymbol{x}, t_l)$
14                     interpolate minimum value of $\Phi^*_{\min} \leftarrow \Phi_{\min}(\boldsymbol{x}, t_l)$
15                     use $\tilde{I}^*$ and $\Phi^*_{\min}$ to compute variations $\frac{\delta E_{\text{data}}}{\delta \Phi_0}$ and $\frac{\delta E_{\text{shape}}}{\delta \boldsymbol{\varphi}}$
16                     compute $\boldsymbol{X}_0 = \boldsymbol{\varphi}_{i,j,k,l}(\boldsymbol{X}_{i,j,k}, t_l)$ (see Algorithm 7.1)
17                     backward interpolate and accumulate update
                           $\Delta\Phi_0(\boldsymbol{X}_0) \leftarrow \Delta\Phi_0(\boldsymbol{X}_0) - \nabla_{\Phi_0} E_{\text{data}}$
18                     accumulate update $\Delta\boldsymbol{\alpha} \leftarrow \Delta\boldsymbol{\alpha} - \nabla_{\boldsymbol{\alpha}} E_{\text{data}}$
19                 end for // $i$
20             end for // $j$
21         end for // $k$
22     end for // $l$
23     compute the variation $\frac{\delta E_{\text{data}}}{\delta \Phi_0}$
24     accumulate shape update $\Delta\Phi_0(\boldsymbol{X}_0) \leftarrow \Delta\Phi_0(\boldsymbol{X}_0) - \nabla_{\Phi_0} E_{\text{shape}}$
25     apply updates $\Phi_0 \leftarrow \Phi_0 + \Delta\Phi_0$, $\boldsymbol{\alpha} \leftarrow \boldsymbol{\alpha} + \Delta\boldsymbol{\alpha}$
26     reinitialize $\Phi_0$ using (7.5)
27 until convergence
```









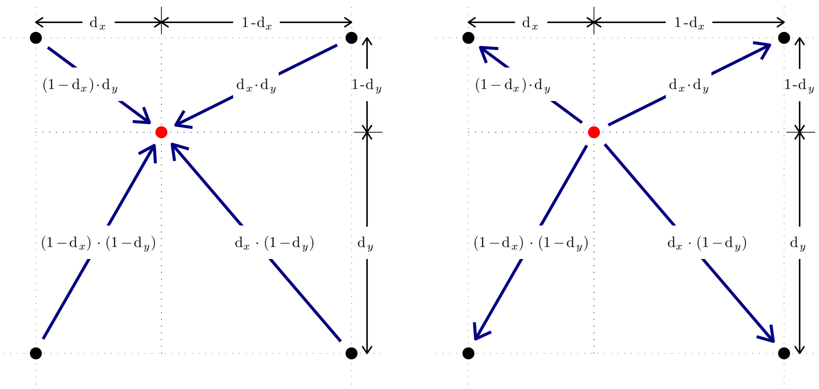
initialized to the *identity transform* by setting all parameters to zero (line 2 in Algorithm 7.2).

Shape regularization

The shape prior (smoothness of the surface) only has to be applied to the reference level set Φ_0 since all other level sets are just warped volumes of the reference reconstruction. This can already be done before iterating over the transformed volumes' voxels. Note, that line 5 requires a subroutine iterating over all reference voxels, though in negligible time.

Warping and ray casting

Lines 7 and 8 are necessary for computing the minimum intensity projections (mIPs) of Φ. Note that instead of first warping the volume and then doing a standard raycasting along straight projection lines, one could also merge the two steps by warping single coordinates / sampling points along the ray. The latter approach has the advantage that only one interpolation takes place. The former approach (as listed in Algorithm 7.2) was favored here since the warped level set volumes Φ are anyways needed for visualization and evaluation purposes of the intermediate reconstruction results.


d_x
$1\text{-}\mathrm{d}_x$
$(1-\mathrm{d}_x)\cdot\mathrm{d}_y$
$\mathrm{d}_x\cdot\mathrm{d}_y$
$1\text{-}\mathrm{d}_y$
$(1-\mathrm{d}_x)\cdot(1-\mathrm{d}_y)$
$\mathrm{d}_x\cdot(1-\mathrm{d}_y)$
d_y
d_x
$1\text{-}\mathrm{d}_x$
$(1-\mathrm{d}_x)\cdot\mathrm{d}_y$
$\mathrm{d}_x\cdot\mathrm{d}_y$
$1\text{-}\mathrm{d}_y$
$(1-\mathrm{d}_x)\cdot(1-\mathrm{d}_y)$
$\mathrm{d}_x\cdot(1-\mathrm{d}_y)$
d_y

(a) Forward interpolation, computing a value at a non-grid location

(b) Backward interpolation, distributing an update from a non-grid location

Figure 7.2: Interpolation of level set updates. (Images reproduced by courtesy of Jakob Vogel [167].)

Updates from data term

The backward interpolation in line 17 is illustrated in Figure 7.2(b). It avoids an inversion of $\boldsymbol{\varphi}$ by applying updates to $\mathit{\Phi}_0$ at non-grid locations $\boldsymbol{X}_0 = \boldsymbol{\varphi}(\boldsymbol{X}_{i,j,k}, t_l)$. This is achieved by distributing $\Delta\mathit{\Phi}_0(\boldsymbol{X}_0)$ to the nearest grid neighbors of $\boldsymbol{X}_0$ using the same weights as for a forward interpolation at $\boldsymbol{X}_0$ (see Figure 7.2(a)). Note that Chen et al. [25] propose an efficient inversion formula for displacement fields that could be used after computing a dense displacement field. Both updates in lines 17 and 18 are performed using the minimum weighting function $\delta\big(d_{M(\boldsymbol{x},t,\mathit{\Phi}_0\circ\boldsymbol{\varphi})}(\boldsymbol{X})\big)$ discussed in Section 7.1. This differs from the updates resulting from the modeling of Kolev and Cremers [92] which uniformly alters all voxels along a ray.

Reinitialization

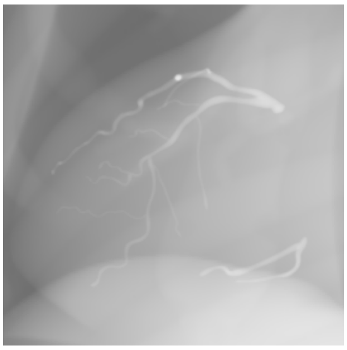
The reinitialization in line 26 could be done using any of the methods described in Section 7.2.2. We decided to follow the standard approach and apply the iterative reinitialization PDE given in (7.5) every few iterations. Although a separate reinitialization has some drawbacks that will be discussed in the next paragraph, it is far more robust than the method given by Li et al. [101, 102] which is numerically unstable for larger step sizes. We generally start with one reinitialization iteration every few iterations of the reconstruction and then gradually switch to one reinitialization per reconstruction iteration when approaching convergence. Thorough reinitializations right from the first iterations would not make any sense and increasing the reinitialization frequency at a later stage still ensures numerical well-behavedness.

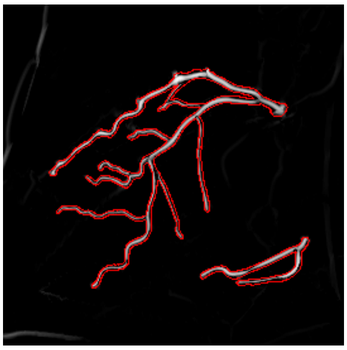
Convergence

Comparing the energy values between subsequent iterations is quite dangerous when sparsely reinitializing the level set function $\mathit{\Phi}_0$ for driving it toward a signed distance function. The reinitialization step does not originate from the energy functional and generally worsens its value (by decreasing the level set function's slope esp. around the interface $\mathit{\Phi}_0 = 0$). Therefore, one has to reinitialize after *every* iteration in order to keep its effect consistent and to be able to do energy comparisons.

Bottlenecks

The computationally most expensive parts of Algorithm 7.2 are lines 7 and 8 taking about 52 % and lines 17 and 18 accounting for about 44 % of the overall computation time, resp. The computation of the mIPs of the warped level set volume would greatly benefit from a more massive parallelization (as on GPUs) enabling a parallel casting of an order of magnitude more rays than on CPUs. The most demanding part of the update accumulations is the distribution of updates to the B-spline coefficients $\boldsymbol{\alpha}$. Line 18 actually contains four nested loops for distributing a voxel's contribution originating from point $(\boldsymbol{X}_{i,j,k}, t_l)$ to all associated spline coefficients. Trading memory for speed, this could be dramatically accelerated using motion vector fields for the deformable components.

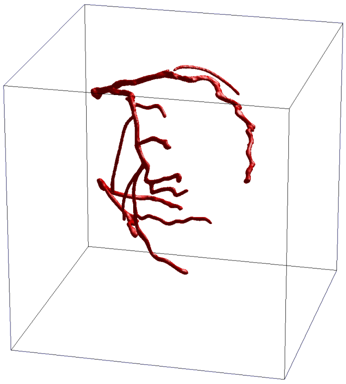


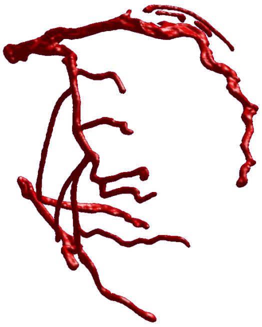


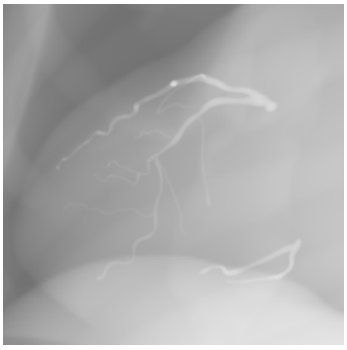
7.4 Hardware Selection and Runtime Considerations

The development of the novel reconstruction functionals like those presented in Chapter 5 involves a number of iterations between modeling and testing. A lot of prototyping and debugging was necessary while experimenting with various types of functionals. Porting the whole algorithm to the GPU would have accelerated the reconstruction speed by a significant factor. However, sticking with multi-core CPUs and OpenMP [119, 120] provided much better debugging capabilities with acceptable wait times for most of the experiments during development. Only the last step in the series of experiments (to be presented in the next chapter) with full resolution data was challenging considering reconstruction times of a few hours. However, the author could not agree more with Herman [65, p. 122] who states that

> "[. . .] electronic hardware used for calculations is getting cheaper and cheaper at an amazing rate. It is unlikely that an efficacious reconstruction algorithm would for long remain unused solely because of computational considerations."

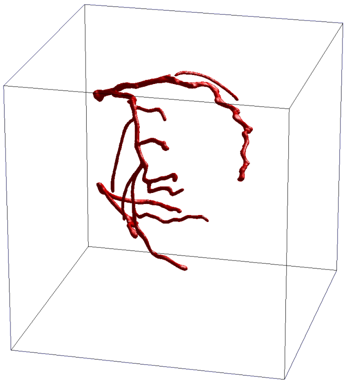


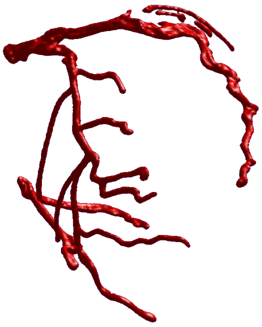


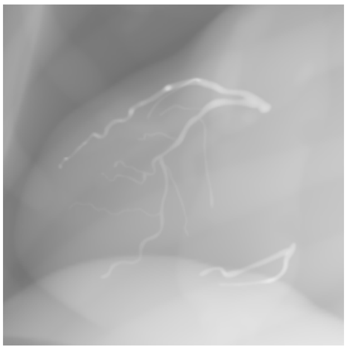


Part III

Evaluation and Discussion







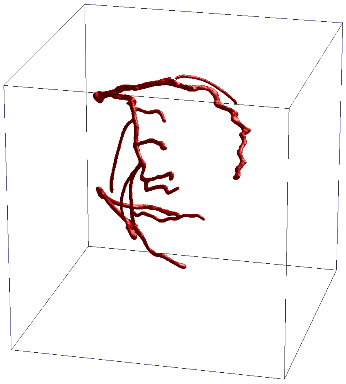
Chapter 8

Experiments

Apart from the experiments we performed for evaluating the two components of our approach (see Chapters 5 and 6), we also tested the combined framework for the reconstruction of dynamic level sets. Increasing the complexity of the problem step by step, we were finally able to obtain promising results from real data.[1]

Before showing the results of these experiments, we will first define some terms that will be used throughout this chapter:

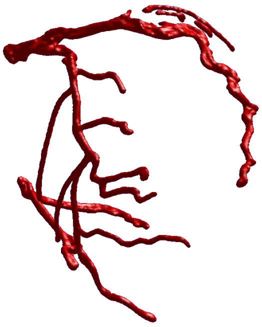
Synthetic data is generated from binary volumetric shapes which are either "painted" into a volume (T) or obtained by segmenting a highly contrasted tomographic volume (S). This volumetric data is then animated using a motion warping function and projected using maximum intensity projections. After adding noise to these MIPs, their pixel values are an approximation to vessel-enhanced angiographic data.



Phantom data is generated using detailed and realistic models for the attenuation and motion of human anatomy – in this case using the XCAT phantom by Segars et al. [145]. Digitally reconstructed radiographs (DRRs) mimicking real X-ray projections are generated by forward projections and are then preprocessed with standard vessel-enhancing filters like Frangi et al. [48], Koller et al. [96], or Krissian et al. [97].

Real data consists of X-ray projections of a living human subject acquired during a rotational angiographic protocol. These projections are also preprocessed using vesselness filters.

Global motion is assumed or modeled non-periodic, i. e., its parameter is the time $t \in \mathcal{T}$.



Periodic motion is assumed or modeled periodic, i. e., its parameter is the heart phase $c \in [0, 1)$. Since 1 is identified with 0, the periodicity condition $\varphi^{(k)}(\cdot, 1) = \varphi^{(k)}(\cdot, 0)$ is required for certain differentiability orders $k \in 0, \ldots, K$.

Rigid motion denotes a dynamic rigid transformation with 6 DOFs for every point in time, modeled with 3 temporally varying Euler angles and 3 temporally varying translation components.

---

[1] Thanks to the support of Tobias Klug and the LRR group at TUM who gave us access to their multi-core system all experiments were performed in reasonable time.

Deformable motion denotes a dynamic deformable transformation with many (theoretically infinite) DOFs for every point in time, modeled with parametrized, varying displacement fields.

This terminology enables us to summarize our experiments as follows: We performed

- $n \times 1\text{D} \rightarrow 2\text{D}$ reconstruction from (orthographic) projections of synthetic objects (see Chapter 5).
- $n \times 2\text{D} \rightarrow 3\text{D}$ reconstruction from (orthographic) projections of synthetic objects (see Chapter 5),
- $n\times 3\text{D} \rightarrow 3\text{D} + t$ segmentation of dynamic CTA data (see Chapter 6), and
- $n \times 2\text{D} \rightarrow 3\text{D} + t$ reconstruction from perspective projections of
    - static synthetic shapes (not discussed here),
    - synthetic shapes with rigid, global motion (presented in Keil et al. [88] and in Section 8.1.1),
    - synthetic shapes with rigid and deformable, periodic motion (presented in Keil et al. [87] and in Section 8.1.2),
    - phantom data with rigid and deformable, periodic motion (XCAT) (see Section 8.2), and
    - real data with rigid and deformable, periodic and global motion (see Section 8.3).

8.1 Synthetic Experiments

Two different synthetic shapes have been created. One was constructed algorithmically by assembling several tubes of varying width in a volume. It is visualized in the top-left image of Figure 8.1 and will be denoted with the letter (T). The other synthetic shape (S) was obtained by segmenting the CT scan of a physical phantom (see lower-left image in Figure 8.1).

These synthetic shapes were warped according to the specific experiment's ground truth motion and then projected with a maximum intensity projection using the calibration matrices of a real C-arm. Gaussian noise (with zero mean and a percentage of the full intensity range) was added in some of the experiments to evaluate the robustness of the reconstruction. Note that there was no preprocessing necessary due to the generation of the input data. The binary projections with the added noise are a simplistic approximation to vessel-enhanced angiographic data.

8.1.1 Rigid, Global Motion

The first set of tests on motion-contaminated data (see also Keil et al. [88]) employed a rigid, non-periodic motion

$$\boldsymbol{\varphi}_{\text{glo}}(\boldsymbol{X}, t, \boldsymbol{\alpha}_{\text{glo}}) = \boldsymbol{R}(t, \boldsymbol{\alpha}_{\text{glo}}) \cdot \boldsymbol{X} + \boldsymbol{T}(t, \boldsymbol{\alpha}_{\text{glo}})$$

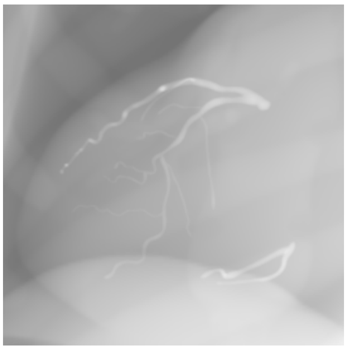
for warping the ground truth volume during MIP generation and for the motion model during reconstruction. The parameters of this setting are summarized in

Table 8.1: Problem dimensions for the synthetic experiment with rigid, non-periodic motions

Projection images:		892 800 pixels[a]
Number of projections	48	
Size	155 px × 120 px	
Pixel spacing on detector	$(2.464\,\frac{\text{mm}}{\text{px}})^2$	
Time per frame	66.8 ms (equals 15 fps)	
Total time	3.2 s	
Angular increment	4° per image, 188° in total	
Reconstruction volume:		125 000 voxels
Size	$(50\,\text{vx})^3$	
Spacing / resolution	$(3\,\frac{\text{mm}}{\text{vx}})^3$	
Motion parametrization[b]:		
Global motion (t):		
Rigid component	12 × 6	72 DOFs
		72 DOFs

[a] Note that the phantom is projected onto a subrectangle covering only approximately 70 % of the projection images, so that the effective number of pixels reduces to approximately 600 000 pixels.

[b] We used quadratic splines for all motions. The first factor gives the DOFs in the temporal dimension, the next three factors – if present – the spatial DOFs, and the final factor equals the number of components to be modeled (6 for rigid motion parameters and 3 for displacements).



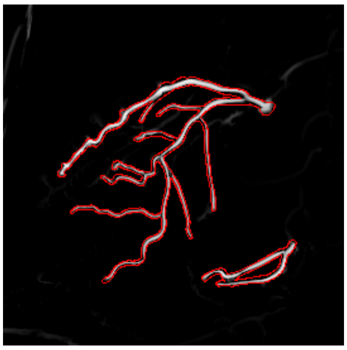
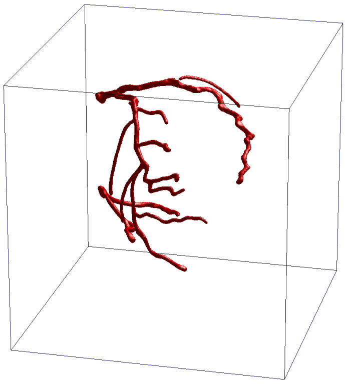
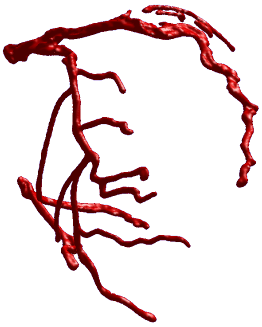


Table 8.1. Note that the reconstruction volume's resolution was actually two to four times coarser than that of the ground truth shapes (T) and (S), resp. This resulted in omissions of the fine structures in the reconstructions of (S) (see bottom row in Figure 8.1). Therefore, this was not a reconstruction error but inherent to the discretization. But since the goal of this first experiment was the evaluation of motion reconstruction errors, it did not influence the results given in Table 8.2. These positional error measurements were obtained by computing the error between the reconstructed motion φ and the known ground truth motion $\widehat{\varphi}$. This is done for every point $\boldsymbol{X}$ in the ground truth's vasculature by warping it to the first frame ($t = 0$) with φ and $\widehat{\varphi}$ and measuring the distance

$$\epsilon(\boldsymbol{X}, t) = \left\| \varphi^{-1}\big(\varphi(\boldsymbol{X}, t), 0\big) - \widehat{\varphi}^{-1}\big(\widehat{\varphi}(\boldsymbol{X}, t), 0\big) \right\|$$

resulting in this frame. This was easily possible due to the rigid motion offering an explicit inversion for every point in time. The errors $\epsilon(\boldsymbol{X}, t)$ were then averaged over all points $\boldsymbol{X}$ inside the ground truth vasculature and all times t to obtain the values in Table 8.2. Due to the confined motion model, the motion estimation is sub-voxel accurate for moderate noise and still at the same scale as the reconstruction volume's resolution of 3 mm for 50 % noise ($\mu = 0$ and $\sigma = 0.5$ for intensities in $[0, 1]$).





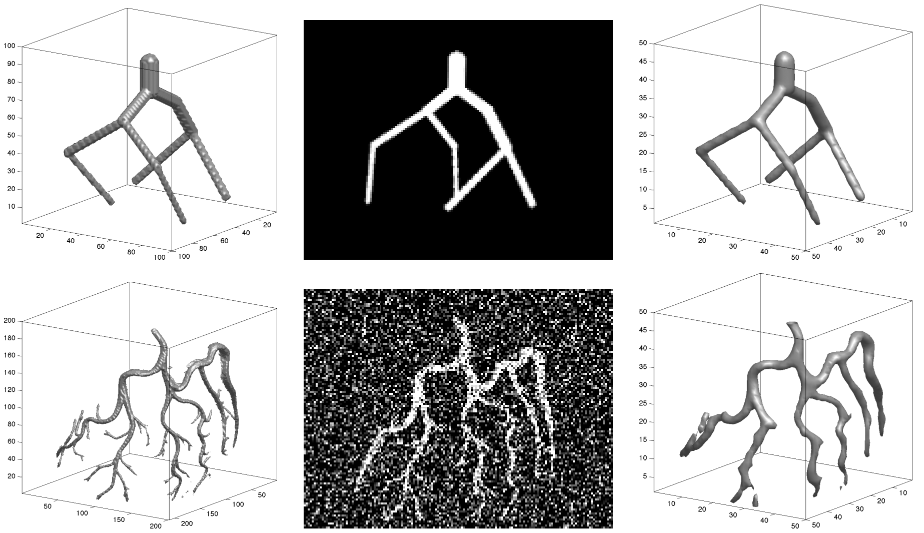

100
90
80
70
60
50
40
30
20
10
20
40
60
80
100
100
80
60
40
20
50
45
40
35
30
25
20
15
10
5
10
20
30
40
50
50
40
30
20
10
200
180
160
140
120
100
80
60
40
20
50
100
150
200
200
150
100
50
50
45
40
35
30
25
20
15
10
5
10
20
30
40
50
50
40
30
20
10

Figure 8.1: Imaging and reconstruction process for synthetic data with global, rigid motion. Top row: Tubular shape (T) without noise. Bottom row: Segmented phantom data (S) with 50 % noise, i. e., $\mu = 0$ and $\sigma = 0.5$ for intensities in $[0, 1]$. From left to right: Ground truth models, example projection, and the final reconstruction. Note that the projections do not show a static setting, but a snapshot of a moving artery model.

Table 8.2: Comparison of motion reconstruction errors. The two data sets shown in Figure 8.1 were reconstructed at three different noise levels. All errors have been evaluated for a series of 10 to 20 experiments.

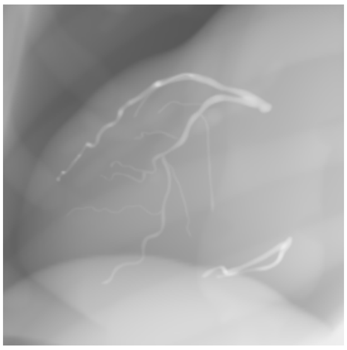
Noise (%)	Mean (mm)	St.D. (mm)	Max. (mm)	Med. (mm)
0	0.81	0.47	4.41	0.72
25	0.83	0.45	4.37	0.75
50	3.68	2.77	11.73	3.07

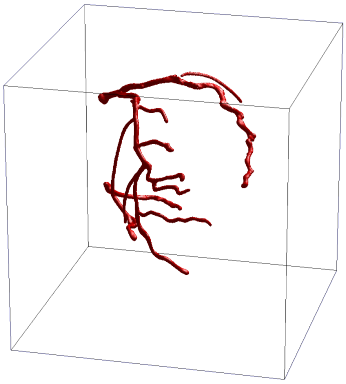
Table 8.3: Problem dimensions for the synthetic experiment with deformable, periodic motions

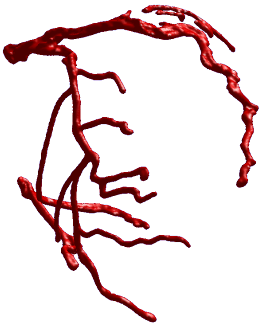
Projection images:		892 800 pixels[a]
Number of projections	48	
Size	$155\,\text{px} \times 120\,\text{px}$	
Pixel spacing on detector	$(2.464\,\frac{\text{mm}}{\text{px}})^2$	
Time per frame	66.8 ms (equals 15 fps)	
Total time	3.2 s	
Angular increment	4° per image, 188° in total	
Heart rate:	∅ 96 bpm	
Reconstruction volume:		125 000 voxels
Size	$(50\,\text{vx})^3$	
Spacing / resolution	$(3\,\frac{\text{mm}}{\text{vx}})^3$	
Motion parametrization[b]:		
Periodic motion (c):		
Rigid component	12×6	72 DOFs
Deformable comp.	$12 \times 5 \times 5 \times 5 \times 3$	4500 DOFs
		4572 DOFs

[a] Note that the phantom is projected onto a subrectangle covering only approximately 70 % of the projection images, so that the effective number of pixels reduces to approximately 600 000 pixels.

[b] We used quadratic splines for all motions. The first factor gives the DOFs in the temporal dimension, the next three factors – if present – the spatial DOFs, and the final factor equals the number of components to be modeled (6 for rigid motion parameters and 3 for displacements).







8.1.2 Deformable, Periodic Motion

The next level in terms of difficulty was the inclusion of deformable motions in addition to the rigid one (see Keil et al. [87]). But this time, the motion

$$\boldsymbol{\varphi}_{\text{per}}\big(\boldsymbol{X}, c(t), \boldsymbol{\alpha}_{\text{per}}\big) = \boldsymbol{R}\big(c(t), \boldsymbol{\alpha}_{\text{per}}\big) \cdot \boldsymbol{X} + \boldsymbol{T}\big(c(t), \boldsymbol{\alpha}_{\text{per}}\big) + \boldsymbol{u}\big(\boldsymbol{X}, c(t), \boldsymbol{\alpha}_{\text{per}}\big)$$

was assumed to be periodic (with the simulated heart phase c). The ground truth volume was warped with translations of up to 10 mm, rotations of up to 10°, and deformations of up to 30 mm. The reconstruction parameters are summarized in Table 8.3.

The main focus of this experiment was an evaluation of the accuracy of the shape reconstruction. A visual comparison (see Figure 8.2) already shows a good alignment of ground truth and reconstruction. A quantitative comparison was obtained using three shape error measures. Using the definitions of the sets illustrated in Figure 8.3, *Dice's coefficient* or the *overlap ratio* is defined as

$$D(\mathrm{R}, \mathrm{G}) = \frac{|\mathrm{G} \cap \mathrm{R}|}{\frac{1}{2}\big(|\mathrm{G}| + |\mathrm{R}|\big)} = \frac{2|\mathrm{G} \cap \mathrm{R}|}{|\mathrm{G}| + |\mathrm{R}|}$$

or, alternatively, as

$$D(\mathrm{R}, \mathrm{G}) = \frac{|\mathrm{TP}|}{\frac{1}{2}(|\mathrm{TP}| + |\mathrm{FP}| + |\mathrm{TP}| + |\mathrm{FN}|)} = \frac{2|\mathrm{TP}|}{2|\mathrm{TP}| + |\mathrm{FP}| + |\mathrm{FN}|} \quad .$$

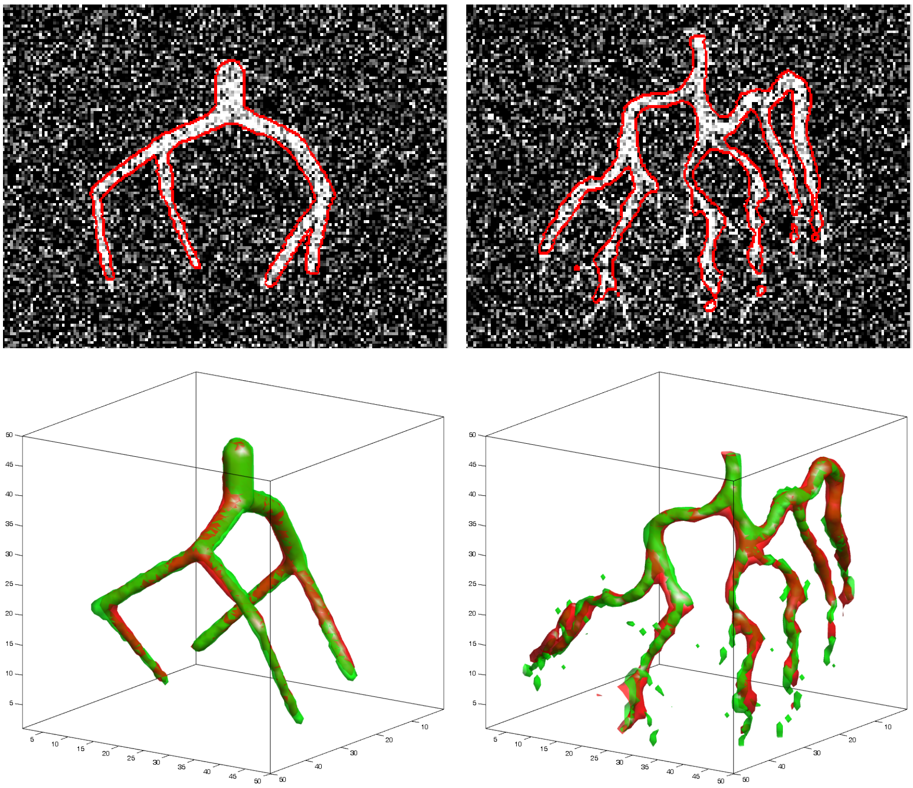


Figure 8.2: Example overlays for evaluation of shape errors for the synthetic data sets with deformable, periodic motion. The noise level for the projections and reconstructions shown is 50 %. Top row: Overlay of reconstructed shape borders (red) on input projection data. Bottom row: 3D overlay of ground truth data (green) and reconstructed shapes (red). Left: Tubular shape (T). Right: Segmented phantom data (S). Note that the whole setup is dynamic and the printed images can only show a snapshot of the non-rigidly moving artery trees.

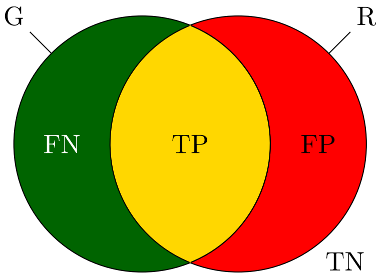


Figure 8.3: Illustration of Dice's similarity coefficient / overlap ratio. G = ground truth, R = reconstruction, TP = true positives, TN = true negatives, FP = false positives, FN = false negatives.

Table 8.4: Comparison of reconstruction errors for two types of synthetic data sets at six different noise levels.

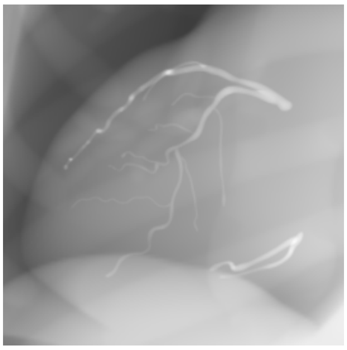
Data set	Noise level (%)	D (%)	S_e (%)	S_p (%)
(T)	0	85.1	86.1	99.9
(T)	10	84.9	84.4	99.9
(T)	20	84.6	83.5	99.9
(T)	30	83.8	80.1	99.9
(T)	40	83.2	80.1	99.9
(T)	50	81.3	75.9	99.9
(S)	0	66.7	75.2	99.6
(S)	10	66.6	78.0	99.6
(S)	20	65.0	73.8	99.6
(S)	30	67.0	74.2	99.6
(S)	40	66.3	72.8	99.6
(S)	50	64.7	71.7	99.6

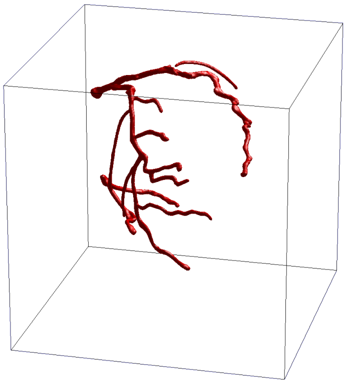
Therefore, D is the ratio of the overlap of two shapes to their average size. In addition to Dice's coefficient, we used the

$$\textit{sensitivity} \quad S_e = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{8.1}$$

$$\text{and the } \textit{specificity} \quad S_p = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \quad . \tag{8.2}$$

Table 8.4 shows that the shape reconstruction is extremely robust with respect to noise. The lower accuracy of the reconstruction of shape (S) is again explained by the coarse resolution of the reference reconstruction's volume. The specificity is of limited interest because it can be pushed arbitrarily close to 100 % by enlarging the reconstruction volume (thereby enlarging TN but usually not FP).





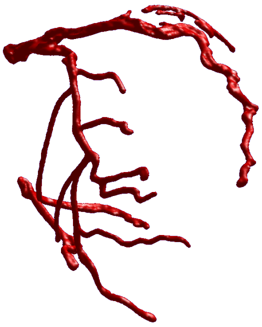


Table 8.5: Problem dimensions for the phantom experiment

Projection images:		123 mil. pixels[a]
Number of projections	133	
Size	$(960\,\mathrm{px})^2$	
Pixel spacing on detector	$(0.32\,\frac{\mathrm{mm}}{\mathrm{px}})^2$	
Time per frame	40 ms (equals 25 fps)	
Total time	5.32 s	
Angular increment	1.5° per image, 198° in total	
Heart rate:	80 bpm	
Reconstruction volume:		7 mil. voxels
Size	$(190\,\mathrm{vx})^3$	
Spacing / resolution	$(0.5\,\frac{\mathrm{mm}}{\mathrm{vx}})^3$	
Motion parametrization[b]:		
Periodic motion (c):		
Rigid component	11×6	66 DOFs
Deformable comp.	$11 \times 7 \times 7 \times 7 \times 3$	11 319 DOFs
		11 385 DOFs

[a] Note that the heart is projected onto a subrectangle covering only approximately 10 % of the projection images, so that the effective number of pixels reduces to approximately 12 million pixels.

[b] We used quadratic splines for all motions. The first factor gives the DOFs in the temporal dimension, the next three factors – if present – the spatial DOFs, and the final factor equals the number of components to be modeled (6 for rigid motion parameters and 3 for displacements).

8.2 Phantom Experiments

The most realistic quantitative evaluations were performed on phantom data derived from the XCAT phantom by Segars et al. [145]. This phantom was built using CT and MR scans of humans and generating spline-based anatomical models from it. Motion models were also derived from real data. The reconstruction parameters are summarized in Table 8.5. This time, the resolution and parameters were all at realistic scales (e. g., $(0.5\,\frac{\mathrm{mm}}{\mathrm{vx}})^3$ for the reconstruction volume).

The whole evaluation was performed using the CAVAREV platform by Rohkohl et al. [138]. It should be noted that although the author of this thesis is also a co-author of the CAVAREV publication, the evaluation reported here was performed completely blind, i. e., without any knowledge of the ground truth data (neither shape nor motion) generated by Rohkohl. In this realistic setting, a prior vessel enhancement was performed for extracting the vascular features from the DRRs.

The reader is referred to the right margin of this thesis for a visualization of this data. The images there are intended to be a flip book[2] with the following movies (from top to bottom): Input DRRs, preprocessed projections with overlaid projected reconstruction outline (in red), 4D reconstruction, and

---

[2] Flip through the pages from front to back. Use the electronic version of this document in an appropriate viewing mode for the best movie experience.

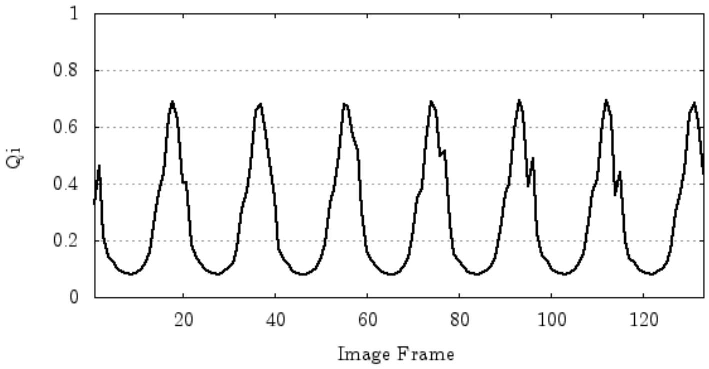
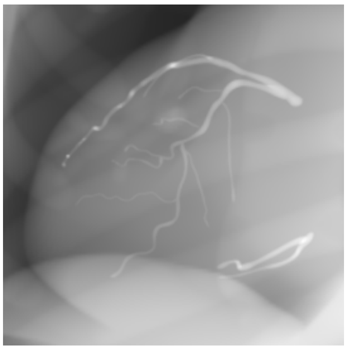
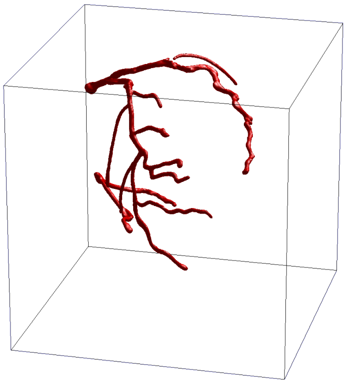
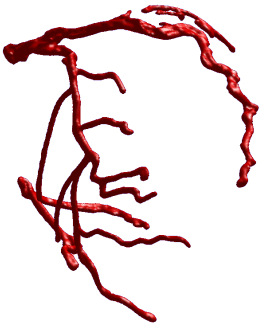


Figure 8.4: Error evaluation of a reconstructed volume for the phantom data set. Comparison with all ground truth volumes. The best overlap score is 69.2 % and peaks occur at frames of similar cardiac phase.







evolution from initialization to final reference shape Φ_0. The two projection sequences have been cropped so that only the relevant, central portions of the images are shown. The reconstructed motion can be best seen by focusing on a specific point (e. g., a bifurcation) in the third movie.

Visual inspection shows that the proximal parts of the LCA are difficult to reconstruct because they are coplanar to the plane of the rotational acquisition. This results in a slow convergence in this region (see bottom movie on the right) and even a small gap in the vasculature.

The two error measures used on the CAVAREV platform are again based on Dice's similarity coefficient. The first measure is a 3D similarity for a reconstructed volume R, obtained by comparing it to all available ground truth volumes G_i (of several motion states) and retaining the maximum overlap ratio

$$Q(\mathrm{R}) = \max_i D(\mathrm{R}, \mathrm{G}_i) \quad .$$

The best overlap achieved across all submitted reconstruction volumes was 69.2 % (see Figure 8.4). A second measure is the 4D overlap

$$Q(\mathrm{R}_1, \ldots, \mathrm{R}_{133}) = \frac{1}{133} \sum_{i=1}^{133} Q(\mathrm{R}_i) \quad ,$$

computed as the average of the 3D errors of all submitted reconstruction frames. Here we achieved 60.5 % (see Figure 8.5). Both errors are within the range of the latest tomographic approaches for cardiac cone beam CT and approx. 10 % better than simple ECG-gated FDK reconstructions according to evaluations with the CAVAREV platform.

8.3 Real Data

Some preliminary experiments with real RA data (courtesy of Prof. Dr. Bachmann and Dr. Rittger, Klinikum Coburg, Germany) were performed without essential modifications compared to the phantom experiments in the

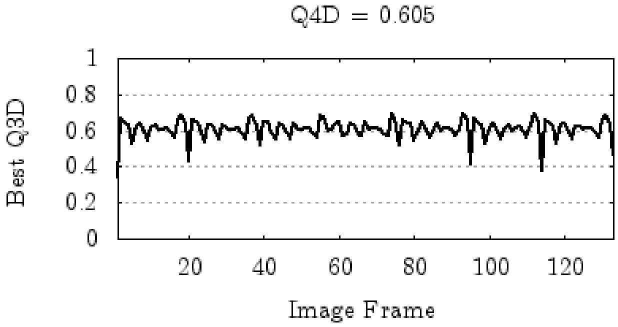
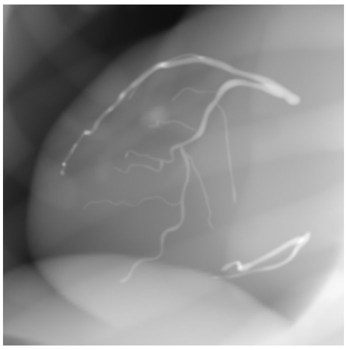
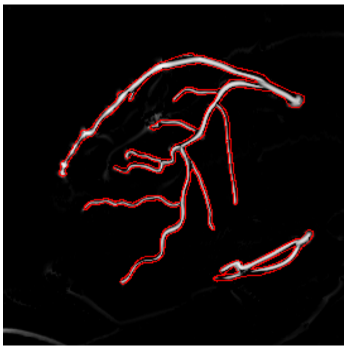


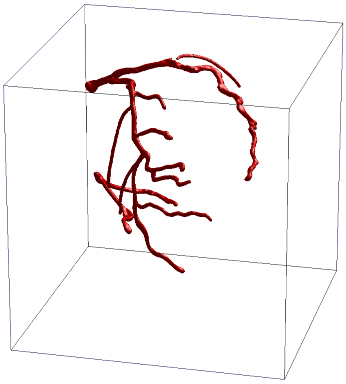
Figure 8.5: 4D error measure for the phantom data set, computed as the average of all best 3D match errors. Average overlap ratio is 60.5 %.

last section. The reconstruction parameters are summarized in Table 8.6 and the results are visualized in Figure 8.6.

The obtained reconstructions were obtained very robustly and the algorithm never diverged. A visual assessment of the accuracy shows that the motion models probably have to be better adapted to realistic heart motions, so that the algorithm is capable of recovering the vasculature from all projections. Furthermore, the contrast injection is not constant over the acquisition time, making some projections misleading due to lacking vessel features. But the integrated solution makes it possible to cope with some level of missing contrast as can be seen from the center-right image of Figure 8.6.







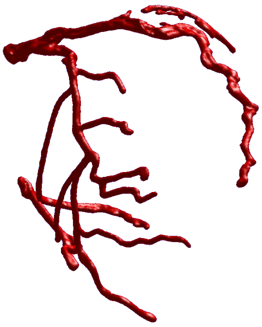


Table 8.6: Problem dimensions for the real experiment

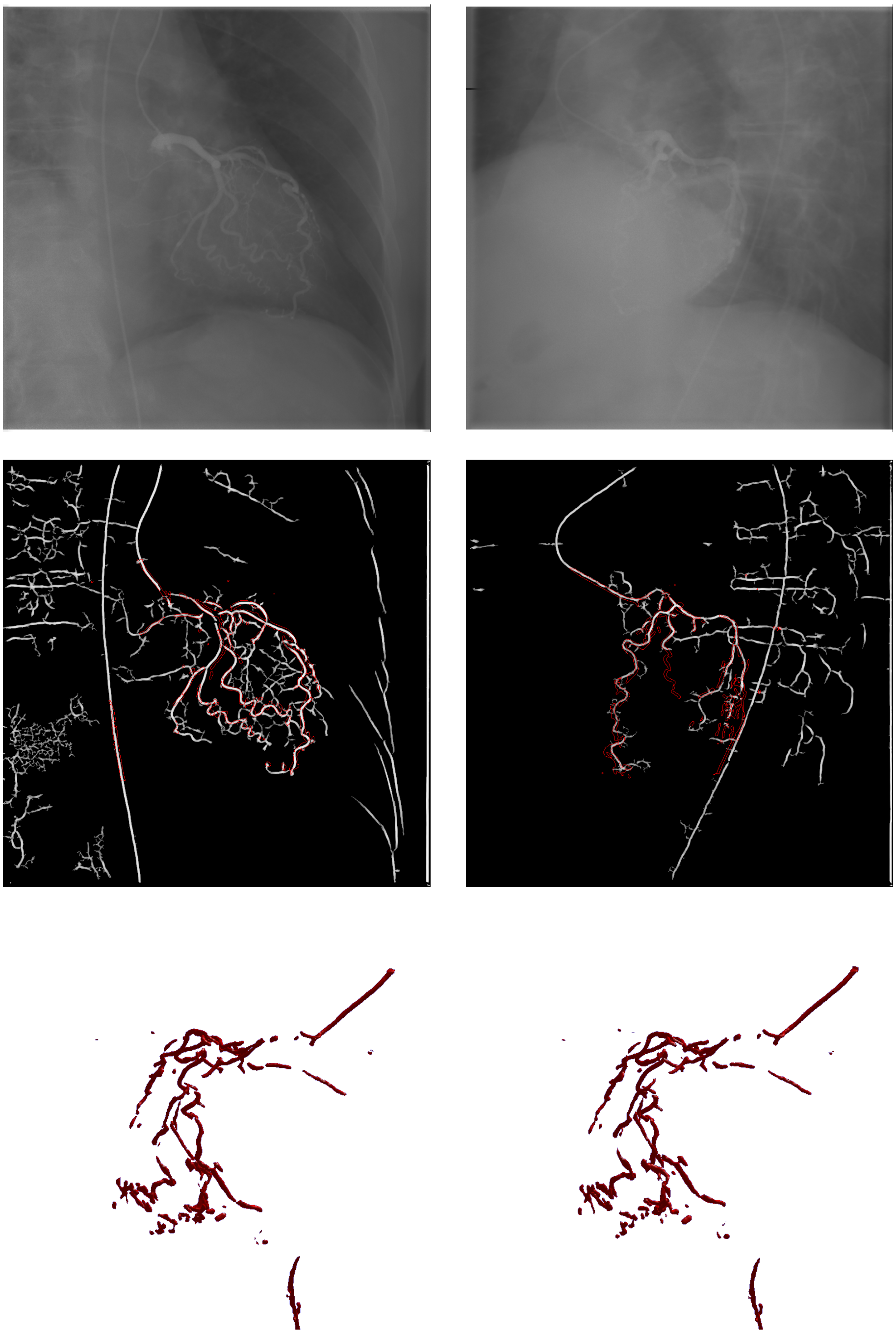
Projection images:		123 mil. pixels[a]
Number of projections	133[b]	
Size	$(960\,\mathrm{px})^2$	
Pixel spacing on detector	$(0.32\,\frac{\mathrm{mm}}{\mathrm{px}})^2$	
Time per frame	∅ 40 ms (equals 25 fps)[c]	
Total time	5.32 s	
Angular increment	1.5° per image, 198° in total	
Heart rate:	∅ 103 bpm	
Reconstruction volume:		8 mil. voxels
Size	$(200\,\mathrm{vx})^3$	
Spacing / resolution	$(0.5\,\frac{\mathrm{mm}}{\mathrm{vx}})^3$	
Motion parametrization[d]:		
Global motion (t):		
Rigid component	12×6	72 DOFs
Deformable comp.	$12 \times 3 \times 3 \times 3 \times 3$	972 DOFs
		1044 DOFs
Periodic motion (c):		
Rigid component	11×6	66 DOFs
Deformable comp.	$11 \times 7 \times 7 \times 7 \times 3$	11 319 DOFs
		11 385 DOFs
		12 429 DOFs

[a] Note that the heart is projected onto a subrectangle covering only approximately 10 % of the projection images, so that the effective number of pixels reduces to approximately 12 million pixels (further reduced by badly contrasted images, see footnote [b]).

[b] Only 103 projections were used for reconstruction due to insufficient contrast dye in the first 30 frames.

[c] In this case these are average times / frame rates. This specific acquisition protocal triggers the X-ray shots equiangular requiring slower frame rates during acceleration and deceleration.

[d] We used quadratic splines for all motions. The first factor gives the DOFs in the temporal dimension, the next three factors – if present – the spatial DOFs, and the final factor equals the number of components to be modeled (6 for rigid motion parameters and 3 for displacements).

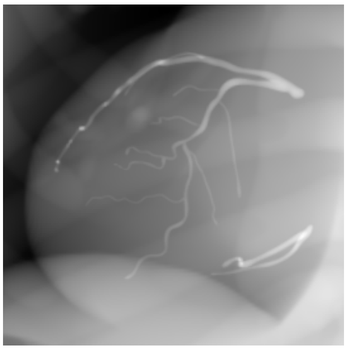


(a)

Figure 8.6: Results for first tests on real angiographic RA data for two acquisition frames (left and right). From top to bottom: Angiograms, vessel-enhanced images with overlaid outline of the reconstruction, 3D rendering of the reconstruction.

Chapter 9

Conclusion

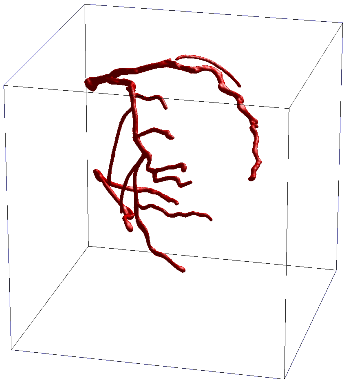


This final chapter wraps up the developments of this thesis and the lessons learned about cardiac C-arm reconstruction as well as dynamic shape models. Based on these experiences, an outlook on potential future developments – for this work and for medical image reconstruction in general – is given.

9.1 Discussion of this Work

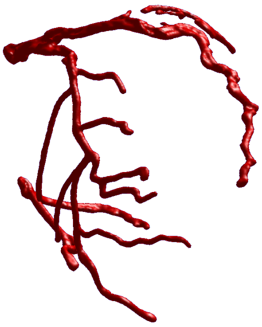
The prior work in the field of cardiac C-arm reconstruction shows that there is currently no consensus on whether successful algorithms should be based on tomographic or symbolic methods. There are basically two streams of work: One seeks to compensate for motion in traditional tomographic methods while the other relies on enhanced motion cues from image features enabling a subsequent motion compensation of tomographic reconstructions.

The models and methods developed in this thesis provide an intermediary between those two worlds by robustly reconstructing shape models defined on dense grids. This required the development of two novel components, namely



- dynamic level sets that can be optimized for shape and motion and
- ray-based silhouette reconstruction functionals tolerating inconsistent projections.

The level set reconstruction proved to yield comparable and in some settings even better results than standard silhouette reconstructions in the experiments performed for testing this component. But their real potential lies in the reconstruction of dynamic shapes from inconsistent projections and thus in combination with the other component – dynamic level sets. In this setting, the projection-based energy functional has the advantage of being more robust to motion than volume-based approaches since they do not depend on the severeness of the motion and thus do not need a dynamic adaptation of parameters like voting thresholds. The dynamic level sets even have potential applications as a stand-alone solution in the dynamic segmentation of 4D data as shown in example experiments. Although not elaborated in this thesis, simultaneous motion extraction and segmentation, e. g., in 4D CT or ultrasound, could benefit from such an integrated shape and motion model enabling the simultaneous usage of all input data and the simple integration of prior knowledge in the form of soft constraints.



Returning to the initial motivation of seeking reconstructions from cardiac C-arm data, these developments were also successfully applied in various experiments, including synthetic, phantom, and real data. Evaluations of the motion and shape reconstruction accuracy using ground truth data (where available) demonstrate the practicability in this advanced setting, even though the versatility of the provided models has not been fully exploited. The discussion on further ideas (see Section 9.2) gives some hints on such potential improvements.

After elaborating on these potential extensions of this work, we will make some reflections on the lessons learned and give an outlook on the future of medical image reconstruction in general.

9.2 Future Work

The models and methods presented in this work have proven to work as intended without further extensions. But depending on the final application and implementation, it makes sense to make further efforts in specific directions.

9.2.1 Algorithm Speed-Up

Leaving the development stage, debugging capabilities are no longer needed and reconstruction speed becomes the main concern. Given this setting, the reconstructions can be accelerated in several ways:

Instead of computing full gradients in every iteration, an SART or randomized ART scheme (see Chapter 2) would likely accelerate convergence the same way these methods accelerate tomographic reconstruction. Instead of processing the input projections sequentially or randomly, one could derive a heuristic method taking not only acquisition angles but also cardiac phase into account for projection selection.

The current algorithm's greatest bottleneck is the usage and esp. the update of the deformable components of the motion models. Instead of using intuitive but computationally expensive tensor product splines one could try out very coarse motion vector fields and interpolate them. This way, some modeling power would be lost, gaining evaluation and update performance (16 relevant spline coefficients instead of 256).

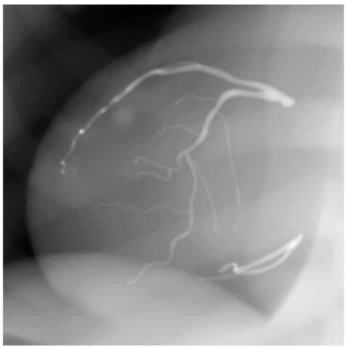
And last but not least, porting the code to GPUs (graphics processing units) is an option. But although widely used in medical imaging research, the graphics card architecture still imposes some limits that might not make them a perfect match for reconstruction (limited memory, reduced speed for arbitrary writes, . . .).

The last two proposals are (as of 2010) mutually exclusive since a full deformation field requires memory in the order of 12 GiB. This is far beyond current graphics cards' memory sizes. So one would either have to wait until such memory sizes become available or use a coarser resolution and interpolation. E. g., a linear interpolation of a displacement field with 1 mm resolution stored for every other image frame would only require about 0.75 GiB of memory.

9.2.2 Modeling Improvements

The models presented so far are still quite general and could be further adopted to the reconstruction of coronary vasculature. This requires some extra work in terms of collecting and evaluating enough data for building statistical priors but could potentially improve the reconstruction results.

Statistical Shape Priors – Tubular Shape Regularization



Statistical shape priors can be smoothly integrated in level set methods (see, e. g., Leventon et al. [100] for early work on this topic and Cremers et al. [36, sec. 8] for a more recent review). Building detailed shape priors, however, is quite difficult and dangerous in medical applications. This is certainly true for coronary trees where variations are significant among the population and ground truth data is hard to obtain.

But since the level set model so far does not include any prior specifically designed for modeling vasculature, the results could even profit from the incorporation of shape priors favoring elongated structures. Gooya et al. [56] propose the use of local correlation matrices of level set normals for this purpose: Assuming that the level set function is a signed distance function with $\frac{\nabla \Phi}{\|\nabla \Phi\|} = 1$, the authors define the local correlation matrix

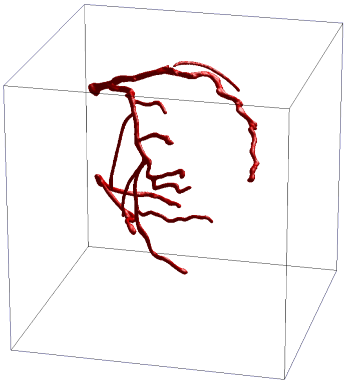
$$\boldsymbol{M}(\boldsymbol{x}) = \int\limits_{B_r(\boldsymbol{X}_0)} H\big(-\Phi(\tilde{\boldsymbol{X}}_{\boldsymbol{0}})\big) \nabla \Phi(\tilde{\boldsymbol{X}}_{\boldsymbol{0}}) \nabla^{\mathsf{T}} \Phi(\tilde{\boldsymbol{X}}_{\boldsymbol{0}}) \; \mathrm{d}\tilde{\boldsymbol{X}}_{\boldsymbol{0}}$$

with $B_r(\boldsymbol{x})$ being a local neighborhood operator for $\boldsymbol{X}_0$. Tubular structures then yield eigenvalues $\lambda_{1,2,3}$ of $\boldsymbol{M}$ with $0 \approx \lambda_1 < \lambda_2 \leq \lambda_3$. This can be favored by adding the shape functional

$$E_{\text{tub}}(\Phi_0) = \int\limits_{\mathcal{V}_0} \delta(\Phi_0) g(\operatorname{tr} \boldsymbol{M}^{-1}) \|\nabla \Phi_0\| \; \mathrm{d}\boldsymbol{X}_0$$

with g being a non-negative decreasing function.

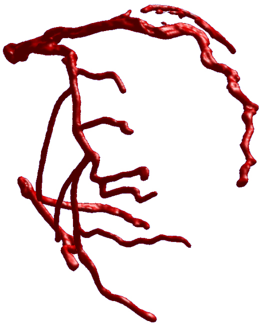
Better Motion Models



Although it is probably not feasible to derive more detailed motion models (regarding arrhythmias etc.), one could still try to optimize the knot distribution of the temporal B-splines. Adjusting the knot density over the cardiac phase c to better match the coronaries' velocities (see Husmann et al. [72]) could improve the motion reconstruction.

Simultaneous Estimation of Cardiac Phase with ECG Signal as a Soft Constraint

Instead of strictly relying on the ECG signal, a simultaneous optimization of $c(t) \in [0, 1)$ (using the measured ECG signal as a weak constraint only) would yield an image-based heart phase estimation. This could be beneficial for arrhythmic patients where the ECG signal may be a flawed indicator (see also Blondel et al. [13] and Rohkohl et al. [135] for image-based detection of cardiac phases).



Adaptive Intensity PDFs

Instead of fixing the intensity distributions p_{hit} and p_{nohit} once during initialization, one could try to adaptively adjust these PDFs by making their parameters (e. g., the mean and variance for Gaussians) part of the optimization. See Chan and Vese [22, 23] for a similar approach in intensity-based segmentation.

9.2.3 Simultaneous Symbolic and Tomographic Reconstruction

The final goal of cardiac C-arm reconstruction may be a tomographic or a symbolic reconstruction. This choice probably mostly depends on the accuracy that can be achieved with symbolic reconstruction. In any case, the two types of reconstruction could benefit from each other if integrated into a common optimization. The chosen Eulerian approach for shape modeling proves to be quite advantageous for such an integration since it allows to easily couple the two (tomographic and symbolic) reconstruction volumes:

Using the notation

$\mathit{\Phi}_0$ for the level set volume (symbolic representation) in the reference frame,
μ_0 for the tomographic volume (of attenuation values) in the reference frame, and
$\boldsymbol{\varphi}$ for the motion,

one may define a composite energy functional

$$\begin{aligned} E(\mathit{\Phi}_0, \mu_0, \boldsymbol{\varphi}) = {} & \underbrace{E_{\mathrm{data}_\mathrm{S}}(\mathit{\Phi}_0, \boldsymbol{\varphi})}_{\text{4D LS reconstr.}} + \underbrace{E_{\mathrm{data}_\mathrm{T}}(\mu_0, \boldsymbol{\varphi})}_{\text{4D tomogr. reconstr.}} + \underbrace{E_{\mathrm{seg}}(\mathit{\Phi}_0, \mu_0)}_{\text{3D CV LS segmentation}} \\ & + E_{\mathrm{prior}_{\mathit{\Phi}_0}}(\mathit{\Phi}_0) + E_{\mathrm{prior}_{\mu_0}}(\mu_0) + E_{\mathrm{prior}_{\boldsymbol{\varphi}}}(\boldsymbol{\varphi}) \quad . \end{aligned} \tag{9.1}$$

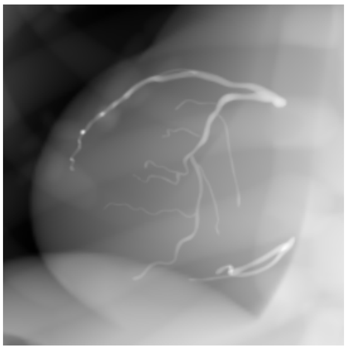
This functional has three new components (compared to a pure symbolic reconstruction as presented in this thesis): $E_{\mathrm{data}_\mathrm{T}}(\mu_0, \boldsymbol{\varphi})$ is the tomographic data term, computing the difference between DRRs and measured line integrals. $E_{\mathrm{seg}}(\mathit{\Phi}_0, \mu_0)$ is a standard segmentation functional (like the intensity-based method by Chan and Vese [22, 23]) relating the symbolic and tomographic volumes. And $E_{\mathrm{prior}_{\mu_0}}(\mu_0)$ is an appropriate regularizer for the tomographic volume, e. g., using the total variation regularization (see, e. g., Rudin et al. [139] and Sidky et al. [157, 158, 156]). (Note that $E_{\mathrm{prior}_{\boldsymbol{\varphi}}}(\boldsymbol{\varphi})$ was just added for completeness in the composite functional (9.1) but the motion's regularity can be obtained by using appropriate explicit models again.)

Optimizing (9.1) using the derivatives $\frac{\delta E}{\delta \mathit{\Phi}_0}$, $\frac{\delta E}{\delta \mu_0}$, and $\frac{\delta E}{\delta \boldsymbol{\varphi}}$ integrates all unknowns and all available information as closely as possible and should give better results compared to sequential approaches.

9.2.4 Reconstruction from Arbitrary Fluoro Sequences

A final improvement, this time in terms of an outlook, would be the reconstruction from fluoroscopic sequences instead of rotational acquisitions. The obvious benefit would be that shots made for navigational purposes could be used for locally improving the reconstruction, too. This, however, would require the robust detection of the contrasted area in order to be able to properly relate the projections obtained at several points during the intervention. In addition, the relative pose between the fluoro runs has to be known (including patient and bed movement) and the calibration has to be valid for any combination of RAO/LAO and cranial/caudal angles. This could be achieved by improved procedures for prior calibration or by the usage of intra-interventional calibration markers (see, e. g., Jain et al. [74] and Navab et al. [117]).

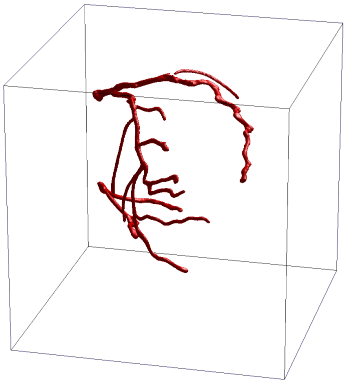
9.3 Conclusions for Cardiac C-Arm Reconstruction



Apart from the arguments given before, there is another, more intuitive one, supporting the incorporation of feature-based methods into cardiac reconstruction: The human mind is capable of gaining a 3D impression of the coronary tree by looking at RA sequences. This is obviously achieved by extracting and tracking vessel features over the image frames. If nature is able to obtain symbolic reconstructions without numerically evaluating line integrals, so should computers. This is not to say that the processing power of computers should not be employed for tomographic reconstruction. But it would certainly help to feed additional shape and motion information into such an algorithm.

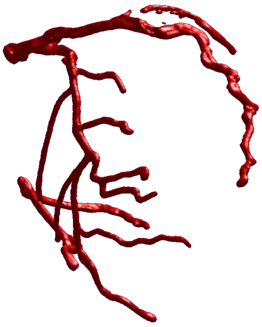
Regarding the models investigated and developed during this work, there were some choices to be made: The "contest" of explicit vs. implicit models resulted in a draw. The former enabled a better incorporation of prior knowledge in the case of motion modeling while the latter turned out to be particularly well-suited for the reconstruction of coronary arteries due to the lack of detailed geometric priors. Level sets also provide the desired soft spatial coupling corresponding quite well to the reality of connected tissue. One could generally argue for explicit models in the case of strong geometric prior knowledge and for implicit models if more flexibility is desired instead.

Finally, it should be noted that although it seems risky to employ new methods in a domain with such well-established solutions as in image reconstruction, the results show that good mathematical modeling should be trusted to yield good algorithms.



9.4 Outlook for Medical Image Reconstruction in General

It does not require visionary skills to predict that medical imaging and surgical procedures or interventions will be further integrated in the future. But a true integration of image reconstruction into existing work flows is naturally the last step when the technology is really mature enough and imaging is merely a *tool* instead of a *process*. Forty years after the invention of CT and after a decade of C-arm CT and cardiac CTA, there are some more developments to be made for seeing cardiac C-arm CT in clinical routine. This specific application as well as other reconstruction problems will be based on the incorporation of more prior knowledge as this allows improvements in several directions: Prior knowledge always reduces the amount of input needed for solving inverse problems. Therefore, it not only allows the solution of problems like cardiac C-arm CT but it also helps to improve existing reconstruction algorithms, e. g., by reducing acquisition time or radiation dose. The usage of total variation regularization (see Rudin et al. [139] and Sidky et al. [157, 158, 156]) is probably only a first step in this direction.



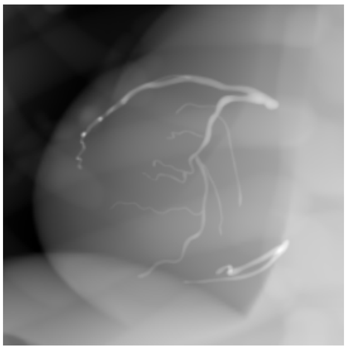
This prompts the conclusion that the age of analytic reconstruction is likely to fade and iterative methods will soon take over. Analytic reconstruction is based on very stringent models and huge efforts have to be made when trying to adapt it to less regularly sampled or otherwise sparse data. Iterative methods, by contrast, are very flexible with respect to the input and allow a much better adaptation to various complex settings.

There are many answers to the question of why analytic methods have been the industry standard for such a long time. (See also Pan et al. [125] for an investigation of this.) The main argument for methods like filtered backprojection are the reduced computational requirements. The early available analytic inversion formulas provided a one-step reconstruction whereas iterative methods required an unacceptably long time in the early days of CT. This head start was further preserved by the availability of efficient numerical schemes like the fast Fourier transform. And the geometric restrictions imposed by analytic formulas were not really a problem.

But current developments in medical imaging will probably yield a shift toward iterative methods. Since the obstacle of computational limits will fade with the advances in computing infrastructure, the algorithms will soon take advantage of the additional flexibility in data requirements and modeling that comes with iterative methods: Acquisition geometries are no longer fixed and irregular paths (e. g., of robotic C-arms) are handled naturally without additional analytic efforts or tweaks. Prior knowledge can be incorporated wherever available, enabling the development of new methods as well as the improvement of existing ones. This is probably the most distinctive feature of iterative methods, that "[...] it is easier to incorporate prior knowledge into the series expansion techniques than into a transform method[1] [...]" (Herman [65, p. 157]). Artifacts may still occur, but not as a direct result of the method employed (as is the case for streak artifacts). Furthermore, they can be handled more effectively in iterative approaches as long as the model (e. g., for beam hardening) is good enough.

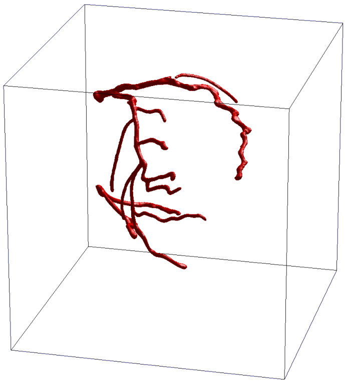
---

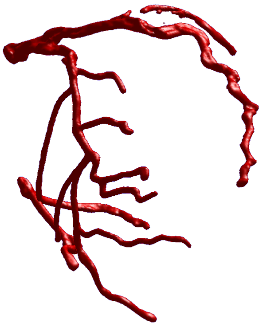
[1] Herman categorizes iterative methods as *series expansion techniques* and analytic reconstruction as *transform method.*



Part IV

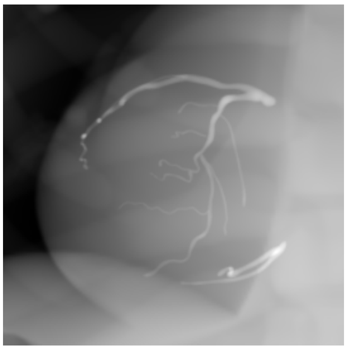
Appendix

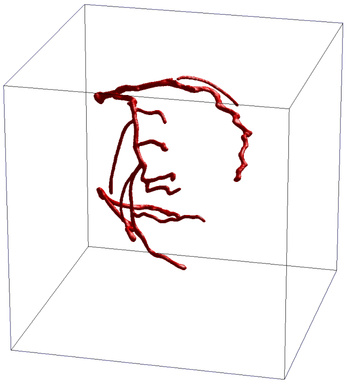




Appendix A

Notation and Nomenclature





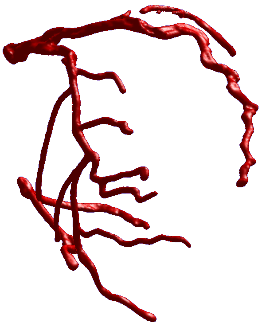


Table A.1: Typefaces and Notation

Notation	Explanation
i, t, E	scalar variables, functions, or functionals
$\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{\varphi}, \boldsymbol{P}$	multidimensional variables or functions (usually $\boldsymbol{x} \in \mathbf{R}^2$ and $\boldsymbol{X} \in \mathbf{R}^3$)
$\mathcal{V}, \mathcal{A}, \mathcal{T}$	sets
s, mm	units like *second* or *millimeter*
$\boldsymbol{x}^\mathsf{T}$	transpose of $\boldsymbol{x}$
$\boldsymbol{x}_\perp$	vector perpendicular to $\boldsymbol{x}$ (rotated by 90° in math. positive direction, i. e., counter-clockwise)
$\langle \boldsymbol{u}, \boldsymbol{v} \rangle$	scalar product of two vectors $\boldsymbol{u}$ and $\boldsymbol{v}$
$\langle f, g \rangle$	inner product of two scalar functions f and g, $\langle f, g \rangle = \int f \cdot \bar{g} \; \mathrm{d}\varOmega$
$\lVert \cdot \rVert$	2-norm for vectors, other norms would be denoted by $\lVert \cdot \rVert_p$
$f(\varOmega)$	image of the set $\varOmega$ under the function f, $f(\varOmega) = \bigcup_{x \in \varOmega} f(x)$
$f^{-1}(y)$	inverse image of y under the function f, $f^{-1}(y) = \{x \colon f(x) = y\}$; note that f does not need to be invertible
$p(A)$	probability of A
$p(A \mid B)$	probability of A, given B
δ	Dirac distribution with $\langle \delta, f \rangle = f(0)$ [1]
$\mathbf{1}_\varOmega$	characteristic function of the set $\varOmega$ with the property $\mathbf{1}_\varOmega(x) = \begin{cases} 1 & : x \in \varOmega \\ 0 & : x \notin \varOmega \end{cases}$
$\frac{\partial f}{\partial x}$, $\partial_x f$	partial derivative of f w. r. t. x
∇f	gradient[2] of f, $\nabla f = \nabla_{\boldsymbol{x}} f = f_{\boldsymbol{x}} = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n} \right)^\mathsf{T}$
div $\boldsymbol{f}$	divergence[2] of $\boldsymbol{f}$, $\operatorname{div} \boldsymbol{f} = \frac{\partial f_1}{\partial x_1} + \dots + \frac{\partial f_n}{\partial x_n}$
Δf	Laplacian[2] of f, $\Delta f = \operatorname{div} \nabla f = \frac{\partial^2 f}{\partial x_1^2} + \dots + \frac{\partial^2 f}{\partial x_n^2}$

continued on next page. . .

---

[1] Compare the italic δ used for the Dirac distribution with the upright $\mathrm{\delta}$ used for variations.

[2] In the context of this work, multidimensional derivative operators are defined w. r. t. the *spatial* arguments of a function (even if the function also depends on time or artificial time).

Table A.1: Typefaces and Notation (continued)

... continued from previous page

Notation	Explanation
H_f	Hessian[2,3], $\mathrm{H}_f = \nabla^2 f = \left(\frac{\partial^2 f}{\partial x_i \partial x_j}\right)_{1\le i,j\le n}$
$\delta_\Phi E\left(\Phi^*;\ \delta\Phi\right)$	variation of E w. r. t. Φ at Φ^* in the direction $\delta\Phi$ [1]
$\frac{\delta E}{\delta\Phi}(\Phi^*)$	variational derivative of E w. r. t. Φ at Φ^* (if the variational derivative exists); the variation in a specific direction $\delta\Phi$ can be calculated as $\delta_\Phi E\left(\Phi^*;\ \delta\Phi\right) = \left\langle \frac{\delta E}{\delta\Phi}(\Phi^*), \delta\Phi \right\rangle$
$\prod_a^b f(x)^{\,\mathrm{d}x}$	continuous product (see Definition B.4)
$\int f(\boldsymbol{\xi})\ \mathrm{dS}(\boldsymbol{\xi})$	surface integral of f
$\boldsymbol{\nu}$	unit outward normal vector field
$\mathcal{C}^k(\Omega)$	set of k times continuously differentiable functions ($k \in \mathbf{N}_0 \cup \infty$) defined on the set Ω, where $\mathcal{C} := \mathcal{C}^0$
$\mathcal{C}_0^k(\Omega)$	set of k times continuously differentiable functions defined on the set Ω which have a compact support in Ω
$\mathcal{L}^p$	set of functions whose absolute value, raised to the p^{th} power ($1 \le p < \infty$), is Lebesgue-integrable
L^p	Banach space of functions from $\mathcal{L}^p$ with the equivalence relation of functions which are equal almost everywhere[4]
$B_r(\boldsymbol{x})$	open ball centered at $\boldsymbol{x}$ with radius r ($B_r(\boldsymbol{x}) := \{\boldsymbol{y} \in \Omega \colon \lVert\boldsymbol{y} - \boldsymbol{x}\rVert < r\}$)
$\partial\Omega$	boundary of the set Ω
$\wp(\mathcal{V})$	power set of $\mathcal{V}$, $\wp(\mathcal{V}) := \{\mathcal{U} \colon \mathcal{U} \subset \mathcal{V}\}$

Table A.2: Variables, Sets, and Functions

Variable		Explanation
$\mathcal{T}$	$\subset \mathbf{R}_0^+$	acquisition time interval
t	$\in \mathcal{T}$	continuous time variable
l	$\in \mathbf{N}_0$	time index or projection number
t_l	$\in \mathcal{T}$	continuous time corresponding to index l
$\mathcal{V}, \mathcal{V}_0$	$\subset \mathbf{R}^3$	reconstruction volumes
$\boldsymbol{X}, \boldsymbol{X}_0$	$\in \mathbf{R}^3$	continuous voxel coordinates in $\mathcal{V}$ and $\mathcal{V}_0$
i, j, k	$\in \mathbf{N}_0$	voxel indexes of the reconstruction grid
$\boldsymbol{X}_{i,j,k,l}$	$\in \mathbf{R}^3$	continuous coordinates corresponding to indexes i, j, k
Φ	$: \mathcal{V} \times \mathcal{T} \to \mathbf{R}$	dynamic/warped level set function
Φ_0	$: \mathcal{V}_0 \to \mathbf{R}$	reference level set function
$\delta\Phi$	$: \mathcal{V} \times \mathcal{T} \to \mathbf{R}$	test function / direction used for expressing variations w. r. t. Φ
$\delta\Phi_0$	$: \mathcal{V}_0 \to \mathbf{R}$	test function / direction used for expressing variations w. r. t. Φ_0

continued on next page...

---

[3] Compare the upright notation of the Hessian H with the italic H for the Heaviside function.

[4] Yes, "almost everywhere" is a precisely defined mathematical term.

Table A.2: Variables, Sets, and Functions (continued)

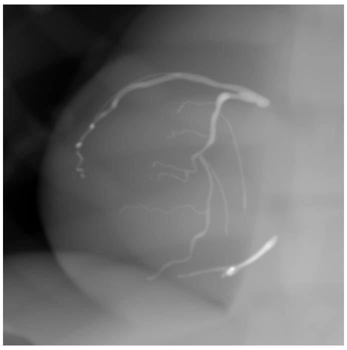
... continued from previous page

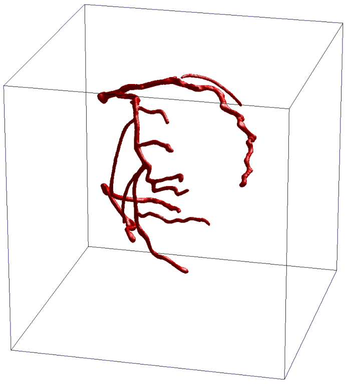
Variable		Explanation
$\boldsymbol{\varphi}$	$: \mathcal{V} \times \mathcal{T} \rightarrow \mathcal{V}_0$	motion function
μ	$: \mathcal{V}_0 \rightarrow \mathbf{R}$	X-ray attenuation volume
$\mathcal{A}$	$\subset \mathbf{R}^2$	projection plane
$\boldsymbol{x}$	$\in \mathcal{A}$	continuous pixel coordinate
r, s	$\in \mathbf{N}_0$	pixel indexes in discrete projection space
$\boldsymbol{x}_{r,s}$	$\in \mathcal{A}$	continuous coordinate corresponding to indexes r, s
I	$: \mathcal{A} \times \mathcal{T} \rightarrow \mathbf{R}$	continuous representation (obtained, e.g., by linear interpolation) of the original X-ray image series
$\tilde{I}$	$: \mathcal{A} \times \mathcal{T} \rightarrow \mathbf{R}$	continuous representation of the preprocessed / vesselness images
E	$: \mathcal{F} \rightarrow \mathbf{R}$	energy function to be minimized ($\mathcal{F}$ is a placeholder for any function space)
$\boldsymbol{P}$	$: \mathcal{V} \times \mathcal{T} \rightarrow \mathcal{A}$	projection operator
$\boldsymbol{P}^{-1}$	$: \mathcal{A} \times \mathcal{T} \rightarrow \wp(\mathcal{V})$	back-projection operator, yielding a set of points in $\mathcal{V}$
τ	$\in \mathbf{R}_0^+$	artificial time variable when formulating a continuous gradient descent (initial values at $\tau = 0$)

Table A.3: Acronyms

Acronym	Explanation
ACS	**a**cute **c**oronary **s**yndrome
ART	**a**lgebraic **r**econstruction **t**echnique
CAD	**c**oronary **a**rtery **d**isease
CBCT	**c**one **b**eam **CT**
CHD	**c**oronary **h**eart **d**isease
CVD	**c**ardio**v**ascular **d**isease
CPU	**c**entral **p**rocessing **u**nit
CT	**c**omputed **t**omography (or **c**omputerized **t**omography)
CTA	**CT** **a**ngiography
CVD	**c**ardio**v**ascular **d**isease
DOF	**d**egree **o**f **f**reedom
DICOM	**D**igital **I**maging and **Co**mmunications in **M**edicine, a format standard for medical imaging data
DRR	**d**igitally **r**econstructed **r**adiograph
ECG	**e**lectro**c**ardio**g**ram
EM	**e**xpectation **m**aximization
FBP	**f**iltered **b**ack **p**rojection

continued on next page...





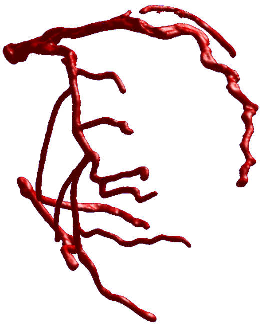


Table A.3: Acronyms (continued)

Acronym	Explanation
... continued from previous page	
FDK	initials of **F**eldkamp, **D**avis, and **K**ress, the triplet of authors that developed a "practical cone-beam algorithm" using filtered back projection in their homonymous paper [46]; *FDK* is used for reference thereto
GPU	**g**raphics **p**rocessing **u**nit
HU	**H**ounsfield **u**nits
IVUS	**i**ntra**v**ascular **u**ltra**s**ound
LAO	**l**eft **a**nterior **o**blique, patient's left shoulder faces the detector[5]
LV	**l**eft **v**entricle (pumping oxygenated blood into the aorta)
MIP	**m**aximum **i**ntensity **p**rojection
mIP	**m**inimum **i**ntensity **p**rojection
ML	**m**aximum **l**ikelihood
MPR	**m**ulti-**p**lanar **r**eformatting or **m**ulti**p**lanar **r**econstruction
MRI	**m**agnetic **r**esonance **i**maging, see NMR
MVF	**m**otion **v**ector **f**ield
NMR	**n**uclear **m**agnetic **r**esonance, see MRI
OR	**o**perating **r**oom
PCI	**p**ercutaneous **c**oronary **i**ntervention
PDF	**p**robability **d**ensity **f**unction
PET	**p**ositron **e**mission **t**omography
QCA	**q**uantitative **c**oronary **a**ngiography
RA	**r**otational **a**ngiography
RAO	**r**ight **a**nterior **o**blique, patient's right shoulder faces the detector[5]
RV	**r**ight **v**entricle (pumping deoxygenated blood into the pulmonary artery)
SART	**s**imultaneous **a**lgebraic **r**econstruction **t**echnique
SPECT	**s**ingle **p**hoton **e**mission **c**omputed **t**omography
TEE	**t**rans**e**sophageal **e**chocardiography
TTE	**t**rans**t**horacic **e**chocardiography
XA	**X**-ray **a**ngiography

---

[5] C-arms are usually positioned with the source under the patient and the detector above the patient for fluoroscopic imaging. Although not relevant in terms of measured intensity and line integrals, this helps shielding the patient's more critical organs from the higher radiation that is present close to the X-ray source. In such a setup, *LAO/RAO* can be translated to "C-arm detector is to the left/right (above) the patient."

Table A.4: Names and Abbreviations of the Coronary Arteries' Branches

Abbreviation	Explanation
LCA	left coronary artery, called *left main artery* up to its first bifurcation
LM	left main artery (LCA up to bifurcation into LAD and LCx)
LAD	left anterior descending artery / anterior interventricular branch of the left coronary artery
LCx	left circumflex artery
OM	left marginal arteries / obtuse marginal branches
RCA	right coronary artery
PIV	posterior interventricular artery, a.k.a. posterior descending artery (PDA); either supplied by the RCA (majority of the population), the LCx, or both

Table A.5: Terminology

Term	Explanation
sagittal	planes dividing the body into a left and a right part
coronal	planes dividing the body into a front and a back part
transverse/axial	planes dividing the body into an upper and a lower part
cranial/superior	closer to the head/top
caudal/inferior	closer to the feet/bottom
proximal	closer to some center or root (usually the body center or another point of interest like the supplying vessel)
distal	more distant from …
systole	contraction of the ventricular myocardium
diastole	relaxation of the ventricular myocardium

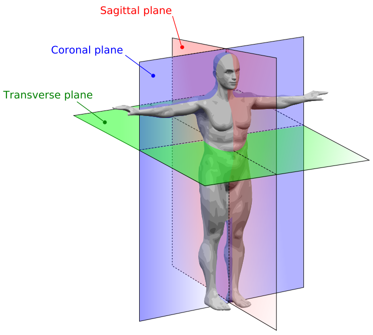
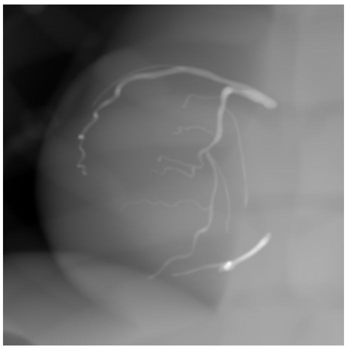
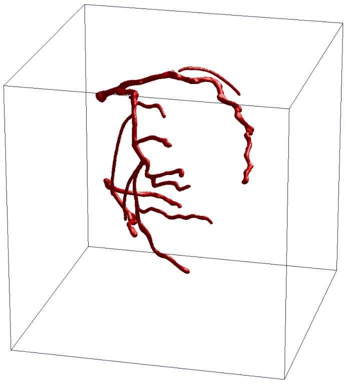
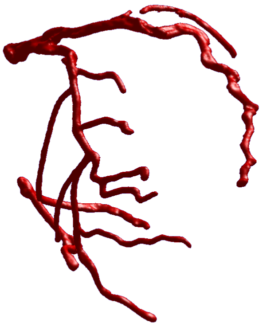


Figure A.1: Planes in human anatomy. (Image source: `http://commons.wikimedia.org/wiki/File:Human_anatomy_planes.svg`. Author: Yassine Mrabet. License: `http://creativecommons.org/licenses/by-sa/3.0/`.)

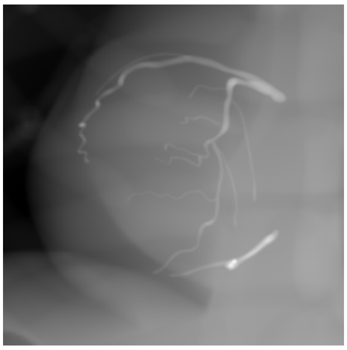


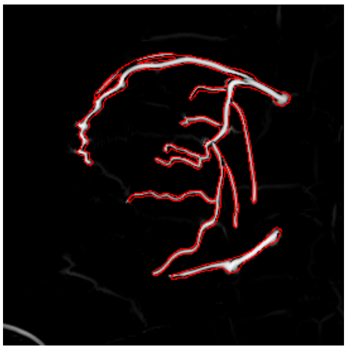


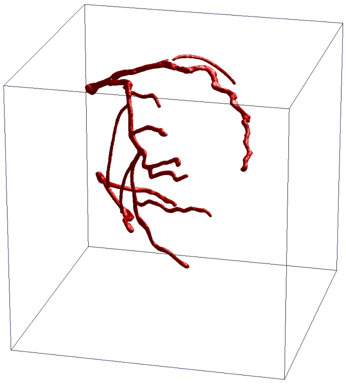


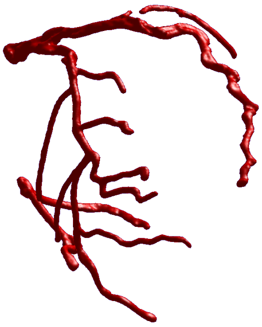
Appendix B

Additional Operators









Definition B.1 (Adjoint operator to the Radon transform in 2D)
The adjoint operator to the Radon transform for functions $g\colon \mathbf{R}^2 \to \mathbf{R}$, $(\alpha, q) \mapsto g(\alpha, q)$ *is defined by*

$$(\mathcal{R}^* g)(\boldsymbol{x}) = \int_0^{2\pi} g\Big(\alpha, \big\langle \boldsymbol{x}, \boldsymbol{n}(\alpha) \big\rangle\Big) \, \mathrm{d}\alpha \quad .$$

The integral in Definition B.1 accumulates all (line integral) values of g for lines $L(\alpha, q)$ that include $\boldsymbol{x}$ by computing the distance q of such lines to the point $\boldsymbol{x}$ as $q = \big\langle \boldsymbol{x}, \boldsymbol{n}(\alpha) \big\rangle$.

Definition B.2 (Hilbert transform)
The Hilbert transform *of a function* $f \in L_2(\mathbf{R})$ *is defined as*

$$\begin{aligned} (\mathcal{H} f)(x) &= \frac{1}{\pi} \ \mathrm{p.\,v.} \int_{\mathbf{R}} \frac{f(y)}{x - y} \, \mathrm{d}y && \text{(B.1a)} \\ &= -\frac{1}{\pi} \ \mathrm{p.\,v.} \int_{\mathbf{R}} \frac{f(x+\tau)}{\tau} \, \mathrm{d}\tau && \text{(B.1b)} \\ &= -\frac{1}{\pi} \lim_{\epsilon \to 0+} \int_{\epsilon}^{\infty} \frac{f(x+\tau) - f(x-\tau)}{\tau} \, \mathrm{d}\tau \quad , && \text{(B.1c)} \end{aligned}$$

*where "*p. v.*" is the* Cauchy principal value *of the improper integral (note the singularity in the denominator).* (B.1b) *is obtained using the substitution* $\tau = y - x$ *and* (B.1c) *by using the definition of the Cauchy principal value (splitting the integral and building the limit).*

Definition B.3 (Fourier transform)
The Fourier transform *of an integrable function* $f \in L_1(\mathbf{R}^n)$ *is defined as*

$$(\mathcal{F} f)(\boldsymbol{\xi}) = \int_{\mathbf{R}^n} f(\boldsymbol{x}) \mathrm{e}^{-2\pi \mathrm{i} \langle \boldsymbol{x}, \boldsymbol{\xi} \rangle} \, \mathrm{d}\boldsymbol{x} \quad .$$

The inverse Fourier transform for $F \in L_1(\mathbf{R}^n)$ *is then given by*

$$(\mathcal{F}^{-1} F)(\boldsymbol{x}) = \int_{\mathbf{R}^n} F(\boldsymbol{\xi}) \mathrm{e}^{2\pi \mathrm{i} \langle \boldsymbol{x}, \boldsymbol{\xi} \rangle} \, \mathrm{d}\boldsymbol{\xi} \quad .$$

The Fourier transform is defined in the *frequency form* in Definition B.3 where $\boldsymbol{\xi}$ is expressed in Hz = s^{-1}. Other definitions based on the *angular frequency* $\boldsymbol{\omega} = 2\pi\boldsymbol{\xi}$ in rad s^{-1} are mostly used in the physics domain and only differ by a multiplicative factor resulting from the parameter transformation.

Definition B.4 (Continuous Product)
Let $f\colon \mathbf{R} \to \mathbf{R}$ *fulfill certain boundedness and positivity conditions on an interval* $[a, b]$*. The continuous product (a.k.a.* mulitplical *or* product integral*) can then be defined in a Riemannian sense as*

$$\prod_a^b f(x)^{\;\mathrm{d}x} = \lim_{n\to\infty} \prod_{i=1}^{n} \Big[f\big(a + i \cdot \Delta x(n)\big) \Big]^{\Delta x(n)}$$

with $\Delta x(n) := \frac{b-a}{n}$ *(if the limit) exists. An alternative definition is derived from the discrete equivalence* $\prod_i f_i = \exp(\sum_i \ln f_i)$ *and is given by*

$$\prod_a^b f(x)^{\;\mathrm{d}x} = \exp\left(\int_a^b \ln f(x)\ \mathrm{d}x \right) \quad .$$

See Frolov [50] for details.

This definition of the continuous product is the continuous equivalent of the indexed product operator and the "product-wise" equivalent to the integration:

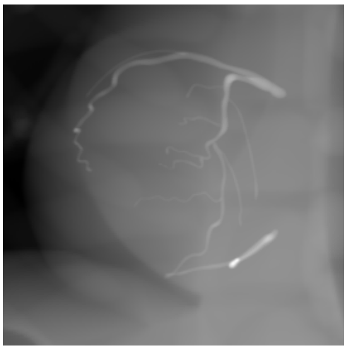
	additive	multiplicative
discrete	$\sum\limits_{i=a}^{b} f(i)$	$\prod\limits_{i=a}^{b} f(i)$
continuous	$\int\limits_a^b f(x)\ \mathrm{d}x$	$\prod\limits_a^b f(x)^{\;\mathrm{d}x}$

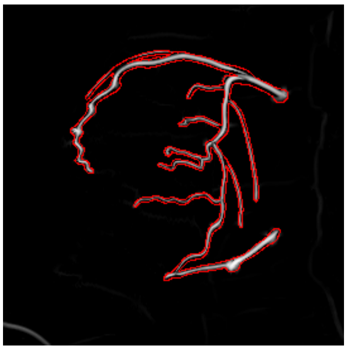
Remark. Note that we have used the same symbol for the continuous as for the discrete product. Other symbols have been suggested and it would be desirable if a distinct symbol emerged in the literature. Until then, $\prod$ should be the most comprehensible symbol.

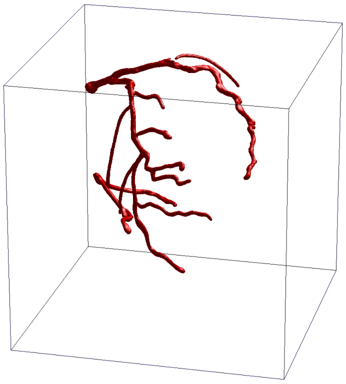
Remark. Volterra's original definition for a continuous product is not used here.

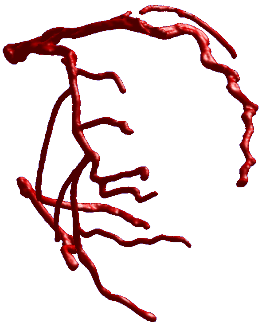
Appendix C

Differentiation and Integration Formulas









This appendix summarizes the differentiation and integration formulas needed in the subsequent appendices as well as in the main part of this document. Proofs are omitted if they are either trivial or if they would go beyond the scope of this document. The reader is pointed to [6, Sec. 2.5.2] and other textbooks for more details.

Proposition C.1 (Derivative of the Norm)
Let $\boldsymbol{x} \in \mathbf{R}^n$ *and*

$$f\colon \begin{cases} \mathbf{R}^n & \to & \mathbf{R} \\ \boldsymbol{x} & \mapsto & \|\boldsymbol{x}\| \end{cases}$$

be the "norm function", then

$$\nabla f(\boldsymbol{x}) = \frac{\boldsymbol{x}}{\|\boldsymbol{x}\|} \quad \forall\, \boldsymbol{x} \neq 0 \quad .$$

Proof.

$$\frac{\partial f}{\partial x_i} = \frac{\partial}{\partial x_i}\left(\sum_{i=1}^{n} x_i^2\right)^{\frac{1}{2}} = \frac{1}{2}\left(\sum_{i=1}^{n} x_i^2\right)^{-\frac{1}{2}} \cdot 2x_i = \frac{x_i}{\|\boldsymbol{x}\|} \qquad \blacksquare$$

Proposition C.2 (Product Rule)
Let $u\colon \mathbf{R}^n \to \mathbf{R}$ *be a scalar field and* $v\colon \mathbf{R}^n \to \mathbf{R}^n$ *a vector field. Then*

$$\operatorname{div}(u \cdot \boldsymbol{v}) = \langle \nabla u, \boldsymbol{v} \rangle + u \cdot \operatorname{div} \boldsymbol{v} \quad .$$

Proof.

$$\begin{aligned} \operatorname{div}(u \cdot \boldsymbol{v}) &= \sum_{i=1}^{n} \frac{\partial (u \cdot \boldsymbol{v})_i}{\partial x_i} = \sum_{i=1}^{n} \frac{\partial (u \cdot v_i)}{\partial x_i} = \sum_{i=1}^{n} \left(\frac{\partial u}{\partial x_i} \cdot v_i + u \cdot \frac{\partial v_i}{\partial x_i}\right) \\ &= \sum_{i=1}^{n} \frac{\partial u}{\partial x_i} \cdot v_i + \sum_{i=1}^{n} u \cdot \frac{\partial v_i}{\partial x_i} = \langle \nabla u, \boldsymbol{v} \rangle + u \cdot \operatorname{div} \boldsymbol{v} \end{aligned} \qquad \blacksquare$$

For the sake of completeness and for showing the similarity of the different integration by parts formulas, we will briefly revisit the integration by parts for functions of a scalar argument.

Proposition C.3 (Integration by parts for functions of scalar arguments) *Let* $f, g\colon [a,b] \subset \mathbf{R} \to \mathbf{R}$*, then*

$$\int\limits_a^b f(x) \cdot g'(x)\ \mathrm{d}x = -\int\limits_a^b f'(x) \cdot g(x)\ \mathrm{d}x + \big[f(x) \cdot g(x)\big]_{x=a}^{b} \quad .$$

Proof. Integrating the product rule

$$[f(x) \cdot g(x)]' = f'(x) \cdot g(x) + f(x) \cdot g'(x)$$

yields the proposition. ■

The equivalent formula in multiple dimensions is

Proposition C.4 (Integration by parts for scalar fields) *Let* $f, g\colon \Omega \subset \mathbf{R}^n \to \mathbf{R}$*. Then*

$$\int\limits_{\Omega} f(\boldsymbol{x}) \cdot \partial_{x_i} g(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} = -\int\limits_{\Omega} \partial_{x_i} f(\boldsymbol{x}) \cdot g(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} + \int\limits_{\partial\Omega} f(\boldsymbol{\xi}) g(\boldsymbol{\xi}) \nu_i(\boldsymbol{\xi})\ \mathrm{dS}(\boldsymbol{\xi}) \qquad \forall\, i \in \{1, \ldots, n\} \quad ,$$

where $\boldsymbol{\nu}$ *is the unit outward normal vector at a point* $\boldsymbol{\xi}$ *on* $\partial\Omega$*.*

Proof. Use the Gauss-Green theorem $\int_{\Omega} \partial_{x_i} u(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} = \int_{\partial\Omega} u(\boldsymbol{\xi}) \cdot \nu_i(\boldsymbol{\xi})\ \mathrm{dS}\boldsymbol{\xi}$ (see [40, App. C.2]) with $u = g \cdot f$. ■

Remark. The similarity of Proposition C.3 and Proposition C.4 becomes apparent when rewriting the former as

$$\int\limits_{[a,b]} f(x) g'(x)\ \mathrm{d}x = -\int\limits_{[a,b]} f'(x) g(x)\ \mathrm{d}x + \big[f(x) g(x) \nu(x)\big]_{x=\partial[a,b]} \quad ,$$

where ν is the unit outward normal of the boundary points of the interval $[a,b]$. In this case, the boundary consists of the two points a and b which have the unit outward normals $\nu(a) = -1$ and $\nu(b) = 1$, resp.

Corollary C.5
Let $\boldsymbol{u}\colon \Omega \subset \mathbf{R}^n \to \mathbf{R}^n$ *be a vector field and let* $v\colon \Omega \subset \mathbf{R}^n \to \mathbf{R}$ *be a scalar field. Then*

$$\int\limits_{\Omega} \langle \boldsymbol{u}(\boldsymbol{x}), \nabla v(\boldsymbol{x}) \rangle\ \mathrm{d}\boldsymbol{x} = -\int\limits_{\Omega} \operatorname{div} \boldsymbol{u}(\boldsymbol{x}) \cdot v(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} + \int\limits_{\partial\Omega} \langle \boldsymbol{u}(\boldsymbol{\xi}), \boldsymbol{\nu}(\boldsymbol{\xi}) \rangle \cdot v(\boldsymbol{\xi})\ \mathrm{dS}(\boldsymbol{\xi}) \quad .$$

Proof. Using Proposition C.4 multiple times (with $f = u_i$, $g = v$, $i = 1, \ldots, n$) and summing up all equations yields

$$\sum_{i=1}^{n} \int\limits_{\Omega} u_i(\boldsymbol{x}) \cdot \partial_{x_i} v(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} = -\sum_{i=1}^{n} \int\limits_{\Omega} \partial_{x_i} u_i(\boldsymbol{x}) \cdot v(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} + \sum_{i=1}^{n} \int\limits_{\partial\Omega} u_i(\boldsymbol{\xi}) v(\boldsymbol{\xi}) \nu_i(\boldsymbol{\xi})\ \mathrm{dS}(\boldsymbol{\xi}) \quad .$$

This can be rewritten using inner products and the divergence operator to obtain Corollary C.5. ■

Remark. Setting $\boldsymbol{u} = \nabla w$, with $w \colon \Omega \subset \mathbf{R}^n \to \mathbf{R}$ being a scalar field, yields Green's first identity

$$\int\limits_{\Omega} \nabla w(\boldsymbol{x}) \cdot \nabla v(\boldsymbol{x}) \;\mathrm{d}\boldsymbol{x} = -\int\limits_{\Omega} \operatorname{div} \nabla w(\boldsymbol{x}) \cdot v(\boldsymbol{x}) \;\mathrm{d}\boldsymbol{x} + \int\limits_{\partial\Omega} \langle \nabla w(\boldsymbol{\xi}), \boldsymbol{\nu}(\boldsymbol{\xi}) \rangle \cdot v(\boldsymbol{\xi}) \;\mathrm{dS}(\boldsymbol{\xi}) \quad .$$

Green's other identities can also be derived from Corollary C.5.

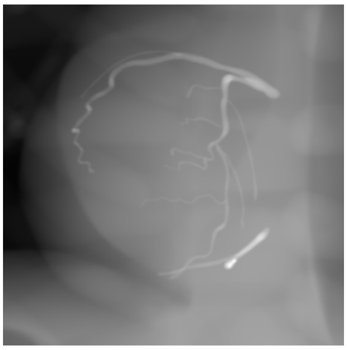
Theorem C.6 (Divergence theorem)

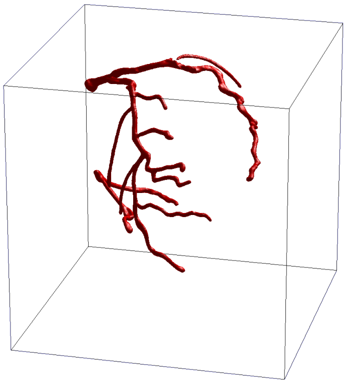
Let $\Omega \subset \mathbf{R}^n$ be compact with a piecewise smooth boundary. If $\boldsymbol{f}$ is a continuously differentiable vector field defined on a neighborhood of Ω, then

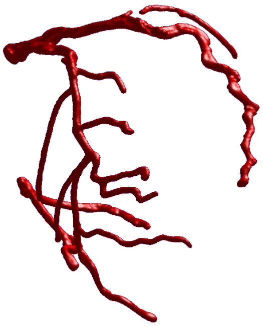
$$\int\limits_{\Omega} \operatorname{div} \boldsymbol{f}(\boldsymbol{x}) \;\mathrm{d}\boldsymbol{x} = \int\limits_{\partial\Omega} \langle \boldsymbol{f}(\boldsymbol{\xi}), \boldsymbol{\nu}(\boldsymbol{\xi}) \rangle \;\mathrm{dS}(\boldsymbol{\xi}) \quad . \tag{C.1}$$

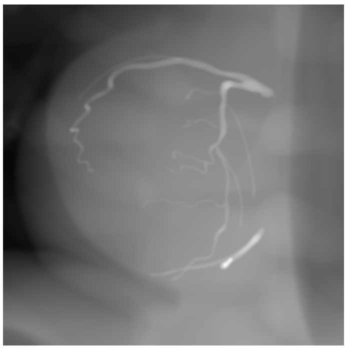
Remark. This theorem is also called *Ostrogradsky-Gauss theorem*, *Gaussian integral theorem*, or simply *Gauss' theorem*.

Proof. Set $\boldsymbol{u} = \boldsymbol{f}$ and $v = 1$ in Corollary C.5 ■









Appendix D

Calculus of Variations

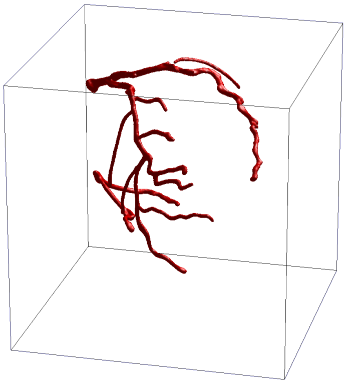
This appendix summarizes the general tools used for calculating update equations by deriving energy terms. The formulas related to level sets are covered in Appendix E.

D.1 Functional Derivative

Definition D.1 (Gâteaux derivative)
Let $\mathcal{X}$ and $\mathcal{Y}$ be locally convex topological vector spaces, $\mathcal{U} \subset \mathcal{X}$ open, and $F\colon \mathcal{X} \to \mathcal{Y}$. The Gâteaux differential of F w. r. t. x at $x^* \in \mathcal{U}$ in the direction $v \in \mathcal{X}$ is defined by

$$\delta_x F\left(x^*;\ v\right) = \lim_{\varepsilon \to 0} \frac{F(x^* + \varepsilon v) - F(x^*)}{\varepsilon} = \frac{\mathrm{d}F}{\mathrm{d}\varepsilon}(x^* + \varepsilon v)\bigg|_{\varepsilon=0}$$

if the limit exists.



Definition D.2 (Functional derivative)
Let $\mathcal{X}$ be a Banach space of functions defined on Ω (e. g., $L^2(\Omega)$) and let

$$E\colon \begin{cases} \mathcal{X} & \to & \mathbf{R} \\ y & \mapsto & E(y) \end{cases}$$

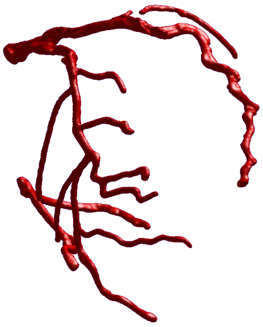
be a functional. For fixed $y^*, \delta y \in \mathcal{X}$

$$\boxed{\delta_y E\left(y^*;\ \delta y\right) = \lim_{\varepsilon \to 0+} \frac{E(y^* + \varepsilon\, \delta y) - E(y^*)}{\varepsilon} = \frac{\mathrm{d}}{\mathrm{d}\varepsilon} E(y^* + \varepsilon\, \delta y)\bigg|_{\varepsilon=0+}}$$

is the functional derivative *or* Gâteaux derivative *of E at y^* in direction δy (if the limit exists). If the Gâteaux derivative exists for* all *variations $\delta y \in \mathcal{X}$, then E is said to be* Gâteaux-differentiable. *And if there exists a function $w \in \mathcal{X}$ such that*

$$\delta_y E\left(y^*;\ \delta y\right) = \langle w, \delta y \rangle_{L^2} = \int_{\Omega} w\, \delta y \ \mathrm{d}\Omega \quad \forall\, \delta y \in \mathcal{X}$$

then E is said to have the Gâteaux derivative $\frac{\delta E}{\delta y}(y^*) = w$ at y^*. If this derivative is bounded (in addition to being linear), then it fulfills the even stronger requirements of a Fréchet derivative.



Corollary D.3 (Derivative of integral functionals)
Now assume that the functional $E(y)$ *is an integral of some function* $f\colon \mathbf{R}^n \times \mathbf{R} \times \mathbf{R}^n \to \mathbf{R}$ *(taking values of* y *and* ∇y *as arguments), i. e.*

$$E(y) = \int\limits_{\Omega} f\big(\boldsymbol{x}, y(\boldsymbol{x}), \nabla y(\boldsymbol{x})\big) \ \mathrm{d}\boldsymbol{x} \quad . \tag{D.1}$$

Applying Definition D.2 yields its functional derivative

$$\delta_y E\left(y^*;\ \delta y\right) = \int\limits_{\Omega} \partial_y f \cdot \delta y + \langle \partial_{\nabla y} f, \nabla \delta y \rangle \ \mathrm{d}\boldsymbol{x} \quad . \tag{D.2}$$

Remark. Note that $\partial_y f$ and $\partial_{\nabla y} f$ in (D.2) are the usual partial derivatives of the function f w. r. t. the arguments y and ∇y, resp. In this case, y and ∇y are just *names* for the arguments of f even though they suggest that they represent functions. f then takes the *values* of the *functions* y and ∇y as actual parameters in (D.1). The double usage of y and ∇y as names of arguments and of functions is tolerated here in order to avoid introducing too many symbols.

We will now turn our attention to calculating the derivative of the minimum functional, a formula that is very likely much less widely known than any other formula used in this work.

Theorem D.4 (Derivative of the minimum functional I)
Let $f \in \mathcal{C}^1(D, \mathbf{R})$, $D \subset \mathbf{R}$ *be a continuously differentiable scalar function. Let* $\mathcal{X} \subset \mathbf{R}^n$ *be bounded and closed and let*

$$E(y) = \min_{\boldsymbol{x} \in \mathcal{X}} f\big(y(\boldsymbol{x})\big)$$

be a functional containing the minimum over $\mathcal{X}$*. The functional derivative of* E *is then*

$$\delta_y E\left(y^*;\ \delta y\right) = \min_{\boldsymbol{x} \in \mathcal{M}} \Big(f'\big(y^*(\boldsymbol{x})\big) \cdot \delta y(\boldsymbol{x})\Big) \quad ,$$

with

$$\mathcal{M} = \arg\min_{\boldsymbol{x} \in \mathcal{X}} f\big(y^*(\boldsymbol{x})\big)$$

being the set of points where the minimum is attained.

Proof. See Girsanov [53, p. 52]. ■

Corollary D.5 (Derivative of the minimum functional II)
Let

$$E(y) = \min_{\boldsymbol{x} \in \mathcal{X}} y(\boldsymbol{x}).$$

be the minimum functional. The functional derivative of E *is then*

$$\delta_y E\left(y^*;\ \delta y\right) = \min_{\boldsymbol{x} \in \mathcal{M}} \delta y(\boldsymbol{x})$$

with

$$\mathcal{M} = \arg\min_{\boldsymbol{x} \in \mathcal{X}} y^*(\boldsymbol{x}).$$

See Figure D.1 for an illustration of Corollary D.5.

Proof. Specializing Theorem D.4 with $f = \mathrm{id}$. ■

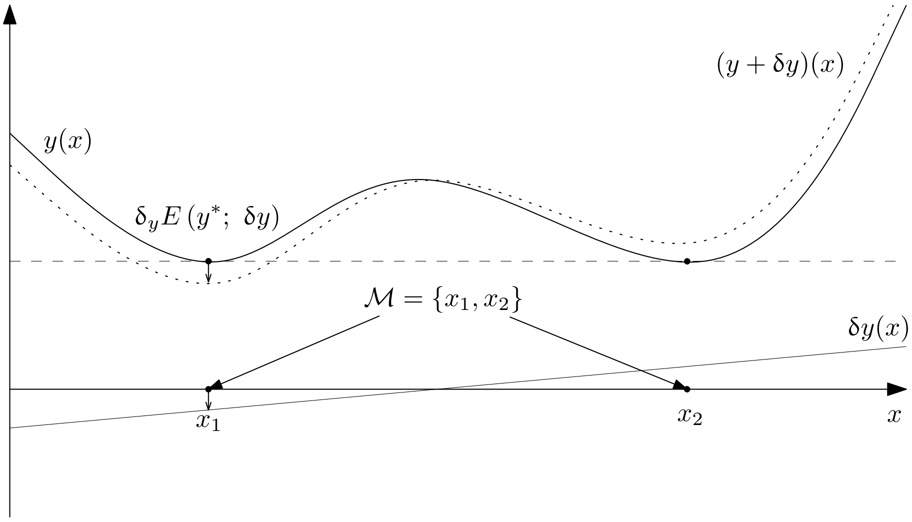
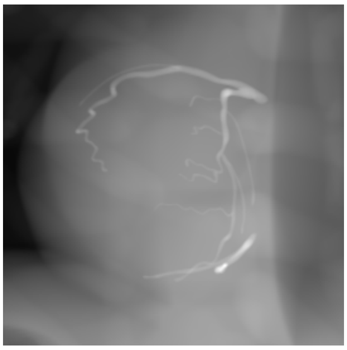
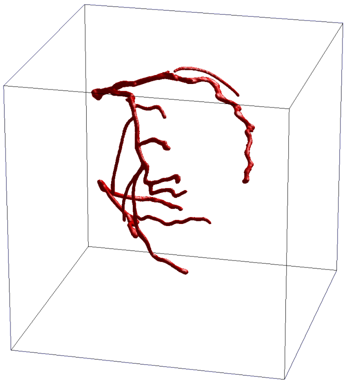
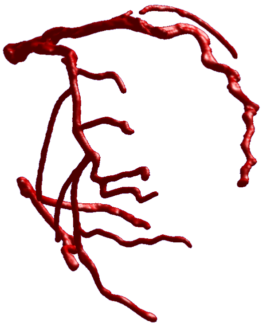


Figure D.1: Illustration of Corollary D.5







D.2 The Fundamental Lemma in the Calculus of Variations

Theorem D.6 (Fundamental lemma in the calculus of variations)
Let $f \in \mathcal{C}(\Omega)$, $\Omega \subset \mathbf{R}^n$, *and* $k \in \mathbf{N}_0 \cup \infty$. *Then*

$$\int\limits_{\Omega} f(\boldsymbol{x}) \cdot \delta y(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} = 0 \quad \forall\, \delta y \in \mathcal{C}_0^k(\Omega) \tag{D.3}$$

is equivalent to

$$f = 0. \tag{D.4}$$

Proof. The implication (D.4) ⇒ (D.3) is obvious. (D.3) ⇒ (D.4) will be proved by contradiction: Let $\boldsymbol{\zeta} \in \Omega$ be a point with $f(\boldsymbol{\zeta}) \neq 0$, without loss of generality $f(\boldsymbol{\zeta}) > 0$. Since $f \in \mathcal{C}(\Omega)$ there exists an $r > 0$ with $f(\boldsymbol{x}) > 0\ \forall\, \boldsymbol{x} \in B(\boldsymbol{\zeta}, r)$. Constructing a function $\delta y \in \mathcal{C}_0^\infty(\Omega) \subset \mathcal{C}_0^k(\Omega)$ with $\delta y(\boldsymbol{x}) > 0\ \forall\, \boldsymbol{x} \in B(\boldsymbol{\zeta}, r)$ and $\delta y(\boldsymbol{x}) = 0\ \forall\, \boldsymbol{x} \in \Omega \setminus B(\boldsymbol{\zeta}, r)$ yields $\int_\Omega f(\boldsymbol{x}) \cdot \delta y(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} > 0$. This is a contradiction to the prerequisite and therefore $f(\boldsymbol{\zeta}) = 0\ \forall\, \boldsymbol{x} \in \Omega$. ■

Since the interesting direction of equivalence in Theorem D.6 is the implication (D.3) ⇒ (D.4) and since $\mathcal{C}_0^\infty \subset \ldots \subset \mathcal{C}_0^1 \subset \mathcal{C}_0$, we now choose the weakest requirement and set $k = \infty$, thereby reducing the set of test functions to $\mathcal{C}_0^\infty(\Omega)$ to obtain

Corollary D.7
Let $f \in \mathcal{C}(\Omega)$, $\Omega \subset \mathbf{R}^n$ *and*

$$\int\limits_{\Omega} f(\boldsymbol{x}) \cdot \delta y(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} = 0 \quad \forall\, \delta y \in \mathcal{C}_0^\infty(\Omega) \quad , \tag{D.5}$$

then

$$f = 0 \quad .$$

Proof. (D.5) implies (D.3) with $k = \infty$ and therefore Theorem D.6 yields $f = 0$. ■

Note that Corollary D.7 only requires test functions from $\mathcal{C}_0^\infty$ which makes it also valid for other sets of test functions, as long as they include $\mathcal{C}_0^\infty$, e. g. test functions from $\mathcal{C}^\infty$.

Remark. The assumptions of Theorem D.6 and Corollary D.7 require f to be continuous. This is only satisfied for continuous energy terms (that implies that the Heaviside function and the Dirac distribution have to be replaced by their mollified approximations, see Appendix E.6). Moreover, the integration by parts performed on the Gâteaux derivative (in order to obtain equation (D.6) in the next section) requires a higher differentiability of the original energy terms.

D.3 Deriving Update Directions Using the Calculus of Variations

Having collected the necessary calculus, we will now derive formulas for reasonable updates to variational optimization problems in the form of

Problem D.8 (Energy minimization)
Let $\Omega \in \mathbf{R}^n$ *and let*

$$E(y) = \int\limits_{\Omega} f\big(\boldsymbol{x}, y(\boldsymbol{x}), \nabla y(\boldsymbol{x})\big) \, \mathrm{d}\boldsymbol{x}$$

be a functional with

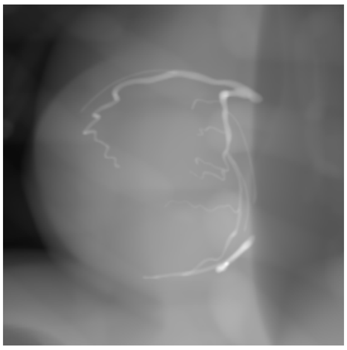
$$f \colon \begin{cases} \mathbf{R}^n \times \mathbf{R} \times \mathbf{R}^n & \to \quad \mathbf{R} \\ \quad (\boldsymbol{x}, y, \nabla y) & \mapsto \quad f(\boldsymbol{x}, y, \nabla y) \end{cases}$$

with enough differentiability (f *continuous w. r. t.* $\boldsymbol{x}$*,* f_y *continuous w. r. t.* $\boldsymbol{x}$*, and* $f_{\nabla y}$ *continuously differentiable w. r. t.* $\boldsymbol{x}$*). We seek to find*

$$\inf_{y \in \mathcal{C}(\Omega)} E(y) \quad .$$

D.3.1 Necessary Conditions for a Stationary Solution

Deriving a necessary condition for a solution to Problem D.8 involves the following steps: After calculating the functional derivative of E w. r. t. y (see



Corollary D.3) and performing an integration by parts in order to replace the factor $\nabla \delta y$ by δy, one obtains a derivative in the form

$$\begin{aligned}\delta_y E\left(y^*;\ \delta y\right) = &\int_\Omega g\big(\boldsymbol{x}, y^*(\boldsymbol{x}), \nabla y^*(\boldsymbol{x}), \mathrm{H}_{y^*}(\boldsymbol{x})\big) \cdot \delta y(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} \\ &+ \int_{\partial\Omega} \Big\langle \boldsymbol{h}\big(\boldsymbol{\xi}, y^*(\boldsymbol{\xi}), \nabla y^*(\boldsymbol{\xi})\big), \boldsymbol{\nu}(\boldsymbol{\xi})\Big\rangle \cdot \delta y(\boldsymbol{\xi})\ \mathrm{dS}(\boldsymbol{\xi}) \quad .\end{aligned} \tag{D.6}$$

Assuming that y^* is a (local) minimizer of E, the condition

$$\delta_y E\left(y^*;\ \delta y\right) = \int_\Omega g \cdot \delta y\ \mathrm{d}\boldsymbol{x} + \int_{\partial\Omega} \langle \boldsymbol{h}, \boldsymbol{\nu}\rangle \cdot \delta y\ \mathrm{dS}(\boldsymbol{\xi}) = 0 \quad \forall\, \delta y \tag{D.7}$$

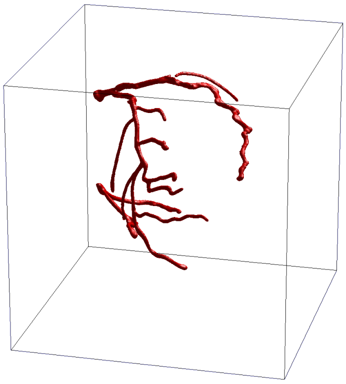
must hold true. Functions δy vanishing on $\partial\Omega$ (e. g. from $\mathcal{C}_0^k(\Omega)$, $k \in \mathbf{N} \cup \infty$) are also allowed in (D.7). The second addend in (D.7) vanishes completely for this type of test functions and therefore

$$\int_\Omega g \cdot \delta y\ \mathrm{d}\boldsymbol{x} = 0$$

must hold true independently for an optimal y^*. By applying Corollary D.7 (assuming a continuous g), this implies that y^* fulfills

$$g\big(\boldsymbol{x}, y^*(\boldsymbol{x}), \nabla y^*(\boldsymbol{x}), \mathrm{H}_{y^*}(\boldsymbol{x})\big) = 0 \quad \forall\, \boldsymbol{x} \in \Omega \quad . \tag{D.8}$$

Allowing arbitrary test functions from $\mathcal{C}^k(\Omega)$ again and inserting the condition (D.8) for an optimal y^* into (D.7) yields



$$\int_{\partial\Omega} \langle \boldsymbol{h}, \boldsymbol{\nu}\rangle \cdot \delta y\ \mathrm{dS}(\boldsymbol{\xi}) = 0 \quad \forall\, \delta y$$

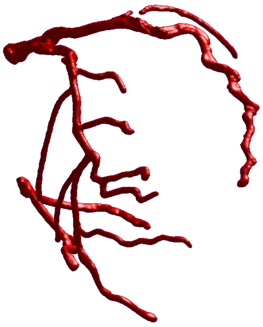
which must also be fulfilled by an extremal y^*. Assuming continuity of $\boldsymbol{h}$, this implies that for every boundary point on $\partial\Omega$ either δy or $\langle \boldsymbol{h}, \boldsymbol{\nu}\rangle$ must vanish[1] for an optimal y^*. This is equivalent to

$$\langle \boldsymbol{h}, \boldsymbol{\nu}\rangle = 0 \quad \forall\, \boldsymbol{\xi} \in \{\boldsymbol{\xi} \in \partial\Omega \colon \delta y(\boldsymbol{\xi}) \neq 0\} \quad .$$

If some boundary parts are fixed, then the test functions δy have to vanish on these parts. This yields no variation. Let $Q \subset \partial\Omega$ be the set of fixed boundary points. Then

$$\begin{aligned} \delta y(\boldsymbol{\xi}) &= 0 \quad \forall\, \boldsymbol{\xi} \in Q \\ \Big\langle \boldsymbol{h}\big(\boldsymbol{\xi}, y^*(\boldsymbol{\xi}), \nabla y^*(\boldsymbol{\xi})\big), \boldsymbol{\nu}(\boldsymbol{\xi})\Big\rangle &= 0 \quad \forall\, \boldsymbol{\xi} \in \partial\Omega \setminus Q \quad .\end{aligned}$$

Together,



$$\boxed{\begin{aligned} g\big(\boldsymbol{x}, y^*(\boldsymbol{x}), \nabla y^*(\boldsymbol{x}), \mathrm{H}_{y^*}(\boldsymbol{x})\big) &= 0 \quad \forall\, \boldsymbol{x} \in \Omega \\ y^*(\boldsymbol{\xi})\ \text{fixed/given} &\quad\ \ \forall\, \boldsymbol{\xi} \in Q \\ \Big\langle \boldsymbol{h}\big(\boldsymbol{\xi}, y^*(\boldsymbol{\xi}), \nabla y^*(\boldsymbol{\xi})\big), \boldsymbol{\nu}(\boldsymbol{\xi})\Big\rangle &= 0 \quad \forall\, \boldsymbol{\xi} \in \partial\Omega \setminus Q \end{aligned}}$$

are the necessary conditions for an extremal y^*.

---

[1] The argument used here is similar to the proof of Theorem D.6. It is also based on the continuity assumptions for the two factors in the integrand as well as a construction of a suitable test function δy for free boundary points.

Remark. Note that the transformation of necessary conditions above is only valid for functionals considered in Problem D.8. Other functionals (e. g., those including a minimum functional) have to be treated differently.

However, computing the stationary solution using the necessary condition for an extremal function y^* is usually not the way to numerically compute a minimizer of Problem D.8. Many authors just state that they perform a gradient descent after computing the functional derivative but this is not at all related to the fundamental lemma or the well-known Euler-Lagrange equation. In fact, it requires to find a descent direction from the directional derivative. One of the few publications correctly stating this fact is Zhao et al. [180, Lemma 2.1]. This will be explained in the following section.

D.3.2 Finding an Update Equation by Steepest Descent

Starting with the functional derivative (D.6), this section justifies the selection of $\frac{\partial y}{\partial \tau} = -g$ as update direction when the stationary point is not reached yet, i. e. $g\big(\boldsymbol{x}, y(\boldsymbol{x}), \nabla y(\boldsymbol{x}), \mathrm{H}_y(\boldsymbol{x})\big) \neq 0$. In a first step, we show that $-g$ is a descent direction (neglecting the boundary integral over h since it's usually automatically fulfilled):

$$\frac{\partial E\big(y(\ldots,\tau)\big)}{\partial \tau} = \delta_y E\left(y;\ \frac{\partial y}{\partial \tau}\right) = \delta_y E\left(y;\ -g(y)\right)$$
$$= \int\limits_{\Omega} g(\boldsymbol{x},\ldots)\cdot[-g(\boldsymbol{x},\ldots)]\ \mathrm{d}\boldsymbol{x} = -\int\limits_{\Omega} \left[g\big(\boldsymbol{x}, y(\boldsymbol{x}), \nabla y(\boldsymbol{x}), \mathrm{H}_y(\boldsymbol{x})\big)\right]^2\ \mathrm{d}\boldsymbol{x} \leq 0$$

In a second step, we will now show that this selection is optimal since it yields the steepest descent: When trying to find a steeper descent direction than $-g$, let ψ be a function which is orthogonal[2] to $-g$, i. e.

$$\int\limits_{\Omega} -g\cdot\psi\ \mathrm{d}\boldsymbol{x} = 0\ \ . \tag{D.9}$$

Setting $\frac{\partial y}{\partial \tau} = -g + \psi$ as new update direction yields

$$\frac{\partial E\big(y(\ldots,\tau)\big)}{\partial \tau} = \delta_y E\left(y;\ \frac{\partial y}{\partial \tau}\right) = \frac{\partial E}{\partial y}(y; -g+\psi) = \int\limits_{\Omega} g\cdot[-g+\psi]\ \mathrm{d}\boldsymbol{x}$$
$$= -\int\limits_{\Omega} g^2\ \mathrm{d}\boldsymbol{x} + \int\limits_{\Omega} g\cdot\psi\ \mathrm{d}\boldsymbol{x} \overset{\text{(D.9)}}{=} -\int\limits_{\Omega} g^2\ \mathrm{d}\boldsymbol{x}$$

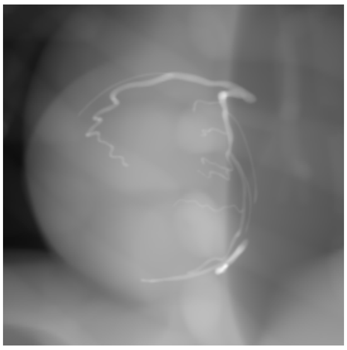
which is no improvement over the choice of

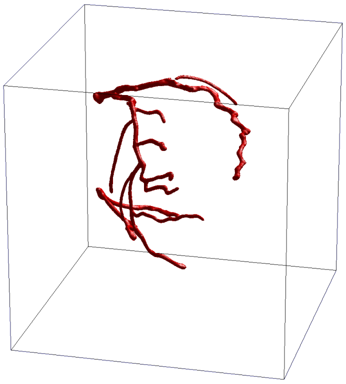
$$\boxed{\frac{\partial y}{\partial \tau} = -g\ \ .}$$

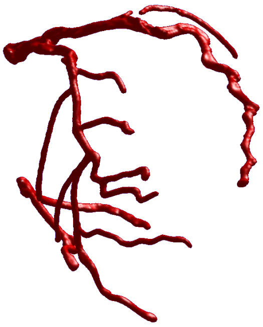
Therefore, $-g$ is the direction of steepest descent.

---

[2] Orthogonality is defined through the inner product which is $\langle u, v\rangle := \int_\Omega u(\boldsymbol{x})\cdot v(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x}$ in $\mathcal{L}^2(\mathbf{R}^n)$.







Appendix E

Calculus for Level Sets

Some calculus that is specific to level sets will be presented in the following sections. For more details, the reader is pointed to Aubert and Kornprobst [6, Sec. 2.4 and 2.5] and Sapiro [140].

E.1 Normal to a Level Set Front

Theorem E.1 (Normal to a level set front)
Let Φ be a level set function defining a front by

$$\Phi(\boldsymbol{x}) \begin{cases} < c & : \boldsymbol{x} \textit{ is inside the front} \\ = c & : \boldsymbol{x} \textit{ is on the front} \\ > c & : \boldsymbol{x} \textit{ is outside the front} \end{cases}$$

and with $c \in \mathbf{R}$. The unit outward normal to the front at a point $\boldsymbol{x}^*$ on the front equals the normed gradient $\frac{\nabla\Phi}{\|\nabla\Phi\|}(\boldsymbol{x}^*)$ of the embedding level set function.

Proof. Let $\mathcal{S} \subset \mathbf{R}^{n-1}$ be open and

$$\boldsymbol{h}\colon \begin{cases} \mathcal{S} & \to & \mathbf{R}^n \\ \boldsymbol{s} & \mapsto & \boldsymbol{h}(\boldsymbol{s}) \end{cases}$$

be a parametric representation of the implicitly defined front with $\boldsymbol{h}(\boldsymbol{s}^*) = \boldsymbol{x}^*$ for some $\boldsymbol{s}^* \in \mathcal{S}$. Since the implicit definition represents the same front as the explicit one,

$$\Phi\big(\boldsymbol{h}(\boldsymbol{s})\big) = c \quad \forall\, \boldsymbol{s} \in \mathcal{S}$$

holds true and therefore (by differentiation w. r. t. s_i)

$$\frac{\partial\Phi\big(\boldsymbol{h}(\boldsymbol{s})\big)}{\partial s_i} = 0 \quad \forall\, \boldsymbol{s} \in \mathcal{S},\ \forall\, i \in \{1,\dots,n-1\} \quad .$$

Evaluating the previous equation at the point $\boldsymbol{s}^*$ yields

$$\begin{aligned} 0 = \frac{\partial\Phi\big(\boldsymbol{h}(\boldsymbol{s}^*)\big)}{\partial s_i} &= \left\langle \nabla\Phi\big(\boldsymbol{h}(\boldsymbol{s}^*)\big), \frac{\partial\boldsymbol{h}}{\partial s_i}(\boldsymbol{s}^*) \right\rangle \\ &= \left\langle \nabla\Phi(\boldsymbol{x}^*), \frac{\partial\boldsymbol{h}}{\partial s_i}(\boldsymbol{s}^*) \right\rangle \quad \forall\, i \in \{1,\dots,n-1\} \quad , \end{aligned}$$

which means that $\nabla\Phi(\boldsymbol{x}^*)$ is perpendicular to all tangents of $\boldsymbol{h}$ in $\boldsymbol{s}^*$ so that $\frac{\nabla\Phi}{\|\nabla\Phi\|}(\boldsymbol{x}^*)$ is the unit normal to $\boldsymbol{h}$ in $\boldsymbol{s}^*$. Since Φ was defined so that it increases in outward direction, $\frac{\nabla\Phi}{\|\nabla\Phi\|}(\boldsymbol{x}^*)$ is the unit outward normal. ∎

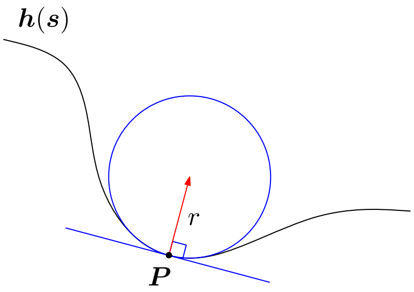


Figure E.1: Osculating circle

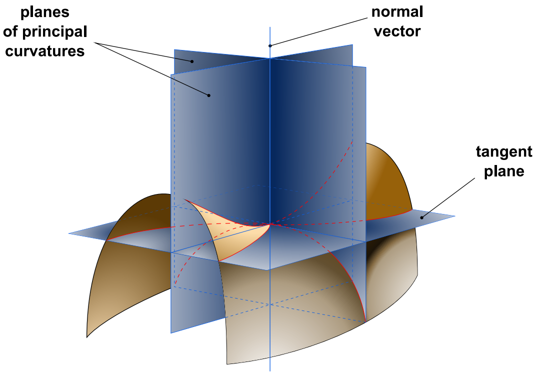


Figure E.2: Planes and curves of principal curvature of a surface point. (Image source: `http://commons.wikimedia.org/wiki/File:Minimal_surface_curvature_planes-en.svg`. Author: Eric Gaba. License: `http://creativecommons.org/licenses/by-sa/3.0/`.)

E.2 Curvature

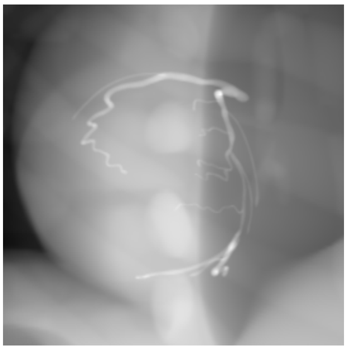
A curve in 2D has just one unambiguously defined curvature, usually denoted κ. It is the inverse of the radius of the osculating circle at a given point (see Figure E.1) and can be computed as $\kappa = \operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right)$ for curves implicitly defined by a level set function Φ.

A surface in 3D has an infinite number of curvatures, defined by the curvatures of all possible plane curves running through a specific point on the surface. The two extrema of these curvatures are called *principal curvatures* of a surface in 3D and are usually denoted by κ_1 and κ_2 (see Figure E.2). From these two principal curvatures, two scalar curvatures are commonly computed, the *Gaussian curvature* $K = \kappa_1\kappa_2$ and the *mean curvature* $H = \frac{1}{2}(\kappa_1 + \kappa_2)$. Only the latter one is of interest in this work. It is computed as $H = \frac{1}{2}\operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right)$ for implicitly defined surfaces in 3D.

Regularizing the surface area of an evolving surface in 3D results in a level set update equation including the expression $\operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right)$ (see Appendix E.4).

The explicit evaluation of this term yields

$$\boxed{\begin{aligned}
\operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right) &= \sum_{i=1}^{3}\frac{\partial}{\partial x_i}\left(\frac{\Phi_{x_i}}{\|\nabla\Phi\|}\right) = \sum_{i=1}^{3}\frac{\|\nabla\Phi\|\Phi_{x_ix_i} - \Phi_{x_i}\sum_{j=1}^{3}\frac{\Phi_{x_j}}{\|\nabla\Phi\|}\Phi_{x_jx_i}}{\|\nabla\Phi\|^2} \\
&= \sum_{i=1}^{3}\frac{\Phi_{x_ix_i}}{\|\nabla\Phi\|} - \sum_{i=1}^{3}\sum_{j=1}^{3}\frac{\Phi_{x_i}\Phi_{x_j}\Phi_{x_ix_j}}{\|\nabla\Phi\|^3} \\
&= \frac{1}{\|\nabla\Phi\|}\left[\sum_{i=1}^{3}\Phi_{x_ix_i} - \sum_{i=1}^{3}\sum_{j=1}^{3}\frac{\Phi_{x_i}}{\|\nabla\Phi\|}\Phi_{x_ix_j}\frac{\Phi_{x_j}}{\|\nabla\Phi\|}\right] \\
&= \frac{1}{\|\nabla\Phi\|}\left[\Delta\,\Phi - \left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right)^{\mathsf{T}}\cdot \mathrm{H}_{\Phi}\cdot\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right)\right] \quad .
\end{aligned}}$$



Remark. After expanding the divergence operator in $\operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right)$ and simplifying terms, one obtains

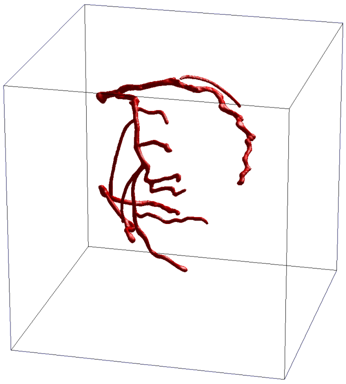
$$\begin{aligned}
\operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right) =& \frac{\Phi_{x_2}^2\Phi_{x_1x_1} + \Phi_{x_3}^2\Phi_{x_1x_1} + \Phi_{x_1}^2\Phi_{x_2x_2} + \Phi_{x_3}^2\Phi_{x_2x_2} + \Phi_{x_1}^2\Phi_{x_3x_3} + \Phi_{x_2}^2\Phi_{x_3x_3}}{\|\nabla\Phi\|^3} \\
&- 2\frac{\Phi_{x_1}\Phi_{x_2}\Phi_{x_1x_2} + \Phi_{x_1}\Phi_{x_3}\Phi_{x_1x_3} + \Phi_{x_2}\Phi_{x_3}\Phi_{x_2x_3}}{\|\nabla\Phi\|^3} \quad .
\end{aligned}$$

This expression, however, has no significant computational advantage.

Remark. The equivalent term in 2D would be

$$\operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right) = \frac{\Phi_{x_2}^2\Phi_{x_1x_1} + \Phi_{x_1}^2\Phi_{x_2x_2} - 2\Phi_{x_1}\Phi_{x_2}\Phi_{x_1x_2}}{\|\nabla\Phi\|^3} \quad .$$

In 2D, this expression equals the curvature κ. (Unlike the infinite number of curvatures of surfaces in 3D, there is only one curvature of 1D curves in 2D, denoted by κ.)



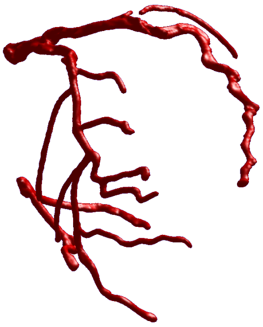
E.3 Integrating a Function Along a Level Set

Theorem E.2 (Integration along a level set)
Let $\Phi\colon \Omega\subset\mathbf{R}^n\to\mathbf{R}$ be a continuously differentiable scalar field. Then

$$\{\Phi = c\} := \{\boldsymbol{x}\in\Omega\colon \Phi(\boldsymbol{x}) = c\} = \Phi^{-1}(c)$$

is the piecewise smooth, $(n-1)$-dimensional hypersurface of the c level set of Φ. Let $f\colon\mathbf{R}\to\mathbf{R}$ be continuously differentiable. Then

$$\boxed{\int\limits_{\{\Phi=c\}} f(\boldsymbol{\xi})\ \mathrm{dS}(\boldsymbol{\xi}) = \int\limits_{\Omega}\delta\big(\Phi(\boldsymbol{x}) - c\big)\cdot\|\nabla\Phi(\boldsymbol{x})\|\cdot f(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} \quad .}$$



Proof. Let $\boldsymbol{n} := \frac{\nabla\Phi}{\|\nabla\Phi\|}$ be the unit outward[1] normal of the oriented level set hypersurface $\{\Phi = c\}$ enclosing the interior $\{\Phi < c\} := \{\boldsymbol{x} \in \Omega \colon \Phi(\boldsymbol{x}) < 0\}$ (see Theorem E.1). Then

$$\begin{aligned}
\int\limits_{\{\Phi=c\}} f \;\mathrm{dS} = \int\limits_{\{\Phi=c\}} \langle f \cdot \boldsymbol{n}, \boldsymbol{n}\rangle \;\mathrm{dS} &\overset{(\mathrm{C.1})}{=} \int\limits_{\{\Phi<c\}} \operatorname{div}\big(f \cdot \boldsymbol{n}\big) \;\mathrm{d}\Omega \\
&= \int\limits_{\Omega} \big(1 - H(\Phi - c)\big) \cdot \operatorname{div}\big(f(\Phi) \cdot \boldsymbol{n}\big) \;\mathrm{d}\Omega \\
&\overset{\text{Corollary C.5}}{=} - \int\limits_{\Omega} \big\langle \nabla\big(1 - H(\Phi - c)\big), f \cdot \boldsymbol{n}\big\rangle \;\mathrm{d}\Omega + \int\limits_{\partial\Omega} \big(1 - H(\Phi - c)\big) \cdot \langle f \cdot \boldsymbol{n}, \boldsymbol{\nu}\rangle \;\mathrm{dS} \\
&\overset{(*)}{=} \int\limits_{\Omega} \left\langle \delta(\Phi - c) \cdot \nabla\Phi, f \frac{\nabla\Phi}{\|\nabla\Phi\|} \right\rangle \;\mathrm{d}\Omega = \int\limits_{\Omega} \delta(\Phi - c) \cdot \|\nabla\Phi\| \cdot f \;\mathrm{d}\Omega \quad .
\end{aligned}$$

Here, we assumed that either $\Phi(\boldsymbol{x}) > c$ or $\langle \boldsymbol{n}, \boldsymbol{\nu}\rangle = 0$ on $\partial\Omega$ in $(*)$. The former condition is fulfilled for all closed hypersurfaces $\{\Phi = c\} \subset \Omega \setminus \partial\Omega$.

Alternatively, one could use the second part of Proposition 3 in Evans and Gariepy [41, Sec. 3.4.4] with $f := \Phi$, $g := f \cdot \|\nabla\Phi\|$, $t = c$, and the property $\{f > t\} = \{H(f - t) = 1\}$ $\int_{\{} f > t\} \cdot \;\mathrm{d}\boldsymbol{x} = \int_\Omega H(f - t) \cdot \;\mathrm{d}\boldsymbol{x}$ to prove the theorem. ■

By integrating over c, Theorem E.2 can be converted to

Corollary E.3 (Coarea formula)
Let $\Phi \colon \Omega \subset \mathbf{R}^n \to \mathbf{R}$ be Lipschitz continuous and assume that for a.e. $c \in \mathbf{R}$ the level set

$$\{\Phi = c\} := \{\boldsymbol{x} \in \Omega \colon \Phi(\boldsymbol{x}) = c\} = \Phi^{-1}(c)$$

is a smooth, $(n-1)$-dimensional hypersurface in Ω. Suppose also that $f \colon \Omega \to \mathbf{R}$ is $\mathcal{L}^n$-integrable. Then

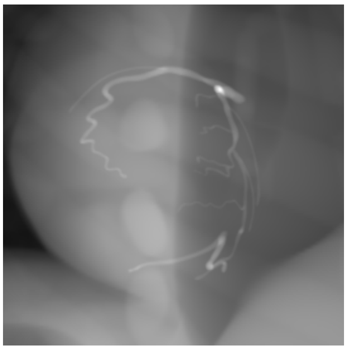
$$\int\limits_{-\infty}^{\infty} \int\limits_{\{\Phi=c\}} f(\boldsymbol{\xi}) \;\mathrm{dS}(\boldsymbol{\xi}) \;\mathrm{d}c = \int\limits_{\Omega} f(\boldsymbol{x}) \|\nabla\Phi(\boldsymbol{x})\| \;\mathrm{d}\boldsymbol{x} \quad ,$$

with s being the arc length parametrization. This formula is called coarea formula *and is pronounced "co-area" (see, e. g., Evans [40, App. C.3] or Aubert and Kornprobst [6, Sec. 2.5.2]).*

Proof. Either integrate Theorem E.2 over c or see Theorem 2 in Evans and Gariepy [41, Sec. 3.4.3] with $f := \Phi$, $g := f$, and $y := c$. ■

---

[1] Here, we are assuming that the level set function decreases towards the interior and increases towards the exterior of the contour $\{\Phi = c\}$. But the results are the same for an antipodal definition.



E.4 Derivative of the Area-minimizing Shape Regularizer

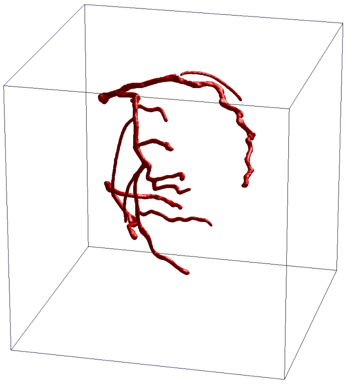
Regularizing a hypersurface is commonly implemented by minimizing its measure (i. e., its curve length or surface area, resp.). The functional

$$E_{\text{shape}}(\Phi) = \int\limits_{\{\Phi=0\}} 1 \ \mathrm{dS} \overset{\text{Theorem E.2}}{=} \int\limits_{\Omega} \delta(\Phi)\|\nabla\Phi\| \ \mathrm{d}\boldsymbol{x}$$

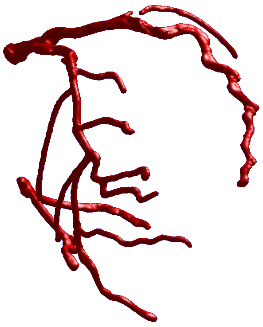
measures this quantity and its derivative

$$\begin{aligned}
\delta_{\Phi} E_{\text{shape}}(\Phi;\ \delta\Phi) &\underset{\text{Proposition C.1}}{\overset{\text{Corollary D.3}}{=}} \int\limits_{\Omega} \delta'(\Phi)\cdot\|\nabla\Phi\|\cdot\delta\Phi + \left\langle \delta(\Phi)\cdot\frac{\nabla\Phi}{\|\nabla\Phi\|}, \nabla\delta\Phi \right\rangle \ \mathrm{d}\boldsymbol{x} \\
&\overset{\text{Corollary C.5}}{=} \int\limits_{\Omega} \delta'(\Phi)\cdot\|\nabla\Phi\|\cdot\delta\Phi \ \mathrm{d}\boldsymbol{x} - \int\limits_{\Omega} \operatorname{div}\left(\delta(\Phi)\cdot\frac{\nabla\Phi}{\|\nabla\Phi\|}\right)\cdot\delta\Phi \ \mathrm{d}\boldsymbol{x} + \\
&\quad + \underbrace{\int\limits_{\partial\Omega} \left\langle \delta(\Phi)\frac{\nabla\Phi}{\|\nabla\Phi\|}, \boldsymbol{\nu} \right\rangle \cdot\delta\Phi \ \mathrm{dS}(\boldsymbol{\xi})}_{=0} \\
&\overset{\text{Proposition C.2}}{=} -\int\limits_{\Omega} \delta(\Phi)\cdot\operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right)\cdot\delta\Phi \ \mathrm{d}\boldsymbol{x}
\end{aligned}$$

yields the so-called *mean curvature motion.* The boundary term $\int_{\partial\Omega}\cdots\ \mathrm{dS}(\boldsymbol{\xi})$ was neglected based on the argument that there is either no variation $\delta\Phi$, no zero level set on $\partial\Omega$, or a vanishing directional derivative of Φ on $\partial\Omega$. In the above form, only the hypersurface at level zero follows this motion because of the factor $\delta(\Phi)$. The usage of a smoothed version of the Dirac distribution (see Appendix E.6) in numerical implementations spreads out the motion to adjacent isolevels of Φ. Furthermore, one may opt for replacing $\delta(\Phi)$ with $\|\nabla\Phi\|$ in order to obtain a geometric motion that is independent of Φ's scaling.



E.5 Derivative of the Signed Distance Regularizer



One possibility to keep the level set function close to a signed distance function according to Li et al. [101, 102] is the penalization of deviations from the property $\|\nabla\Phi\| = 1$ by adding

$$E_{\text{dist}}(\Phi) = \int\limits_{\Omega} \frac{1}{2}\cdot\left(\|\nabla\Phi\| - 1\right)^2 \ \mathrm{d}\boldsymbol{x}$$

to the energy terms.

Using basically the same theorems and assumptions as in Appendix E.4, we obtain

$$\begin{aligned}
\delta_{\Phi} E_{\text{dist}}\left(\Phi;\ \delta\Phi\right) &\overset{\text{Corollary D.3}}{\underset{\text{Proposition C.1}}{=}} \int\limits_{\Omega} \left\langle \left(\|\nabla\Phi\| - 1\right) \cdot \frac{\nabla\Phi}{\|\nabla\Phi\|}, \nabla\delta\Phi \right\rangle \,\mathrm{d}\boldsymbol{x} \\
&= \int\limits_{\Omega} \left\langle \nabla\Phi - \frac{\nabla\Phi}{\|\nabla\Phi\|}, \nabla\delta\Phi \right\rangle \,\mathrm{d}\boldsymbol{x} \\
&\overset{\text{Corollary C.5}}{=} - \int\limits_{\Omega} \operatorname{div}\left(\nabla\Phi - \frac{\nabla\Phi}{\|\nabla\Phi\|}\right) \cdot \delta\Phi \,\mathrm{d}\boldsymbol{x} + \underbrace{\int\limits_{\partial\Omega} \left\langle \nabla\Phi - \frac{\nabla\Phi}{\|\nabla\Phi\|}, \boldsymbol{\nu} \right\rangle \cdot \delta\Phi \,\mathrm{dS}\boldsymbol{\xi}}_{=0} \\
&= \int\limits_{\Omega} \left[-\Delta\Phi + \operatorname{div}\left(\frac{\nabla\Phi}{\|\nabla\Phi\|}\right)\right] \cdot \delta\Phi \,\mathrm{d}\boldsymbol{x} \quad .
\end{aligned}$$

E.6 Mollified Versions of Heaviside Function and Dirac Distribution

Remark. See Zhao et al. [180], Chan and Vese [22, 23] for the contents of this section.

Using mollified[2] versions δ_ϵ and H_ϵ of the Heaviside function H and the Dirac distribution δ (which is the derivative of the Heaviside function) preserves the global character of the update equation (influencing the level set function not only on the front itself) and at the same time also enables a focus on the evolving front by decreasing the smoothness parameter ϵ during later iteration cycles. This choice also determines how far forces are extended to the neighborhood of the evolving front.

Choosing smaller values for ϵ yields sharper approximations and therefore closer resembles the original formulation. However, many terms in the update equations have the form

$$\frac{\partial\Phi}{\partial\tau} = \delta(\Phi) \cdot \ldots$$

Therefore, choosing larger values for ϵ lets the level set evolution act more globally, even for discontinuously defined energy functionals. This has the advantage that the evolution approaches the global optimum faster since the shape may not only evolve at its current borders but also at distant locations. In the case of disconnected parts, a mollified Heaviside function may even be needed for reaching the global optimum. We will now present two common approximations to the Heaviside function and the Dirac distribution, both sharing the properties

$$\lim_{\epsilon\to 0} H_\epsilon = H \tag{E.1}$$

$$\lim_{\epsilon\to 0} \delta_\epsilon = \delta \tag{E.2}$$

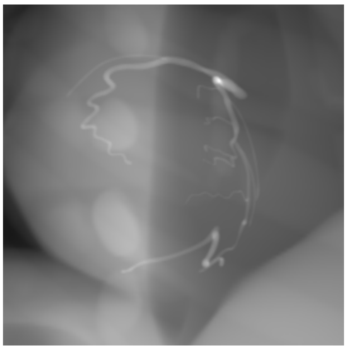
$$H_\epsilon^{\text{loc}}(-x) = 1 - H_\epsilon^{\text{loc}}(x) \quad \Rightarrow \quad H_\epsilon^{\text{loc}}(0) = \frac{1}{2} \quad , \tag{E.3}$$

where (E.2) is to be understood in a distributional sense. The symmetry property (E.3) enables to either use $1 - H(\Phi)$ or $H(-\Phi)$ when defining energies for the "inside part" of the shape defined by the level set Φ.

---

[2] Yes, this word is indeed spelled "mollified" and refers to the tamed characteristics compared to the Dirac functional.

E.6.1 Global Support by the arctan Approximation



Choosing an approximation for the Heaviside function which is smeared out over the whole domain $\mathbf{R}$ results in a Dirac distribution with global a support and therefore ensures the influence of the energy terms on all level sets. One such approximation is

$$\begin{aligned} H_\epsilon^{\mathrm{glo}}(x) &= \frac{1}{2} + \frac{1}{\pi} \cdot \arctan \frac{x\pi}{\epsilon} \\ \Rightarrow \quad \delta_\epsilon^{\mathrm{glo}}(x) &= \frac{\epsilon}{\epsilon^2 + x^2\pi^2} \quad . \end{aligned}$$

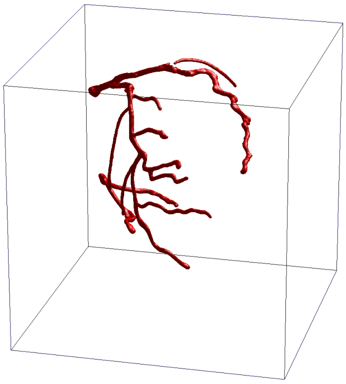
In addition to (E.1)–(E.3) it has the properties

$$\begin{aligned} H_\epsilon^{\mathrm{glo}} &\in \mathcal{C}^\infty(\mathbf{R}) \\ \operatorname{supp} \delta_\epsilon^{\mathrm{glo}} &= \mathbf{R} \\ \delta_\epsilon^{\mathrm{glo}}(0) &= \frac{1}{\epsilon} \quad . \end{aligned}$$

However, since the level set function in this work is initialized with zero and since only one connected component is to be reconstructed, the influence on all level sets is not as advantageous as in other applications but merely increases computational cost.

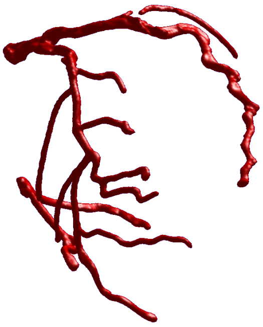
Remark. We slightly modified the version of the global approximation compared to other sources such as Chan and Vese [22, 23]. Doing so yields the same peak value for the both (the global and the local) Dirac approximation for the same parameter ϵ.

E.6.2 Local Support by Piecewise Approximation



Choosing an approximation with a local support enables to reduce the influence to a local neighborhood of the zero level set. This is advantageous when using narrow band methods (see Adalsteinsson and Sethian [2]) or if the restriction to a band is otherwise desirable (e. g., for speeding up the level set updates). Such an approximation is

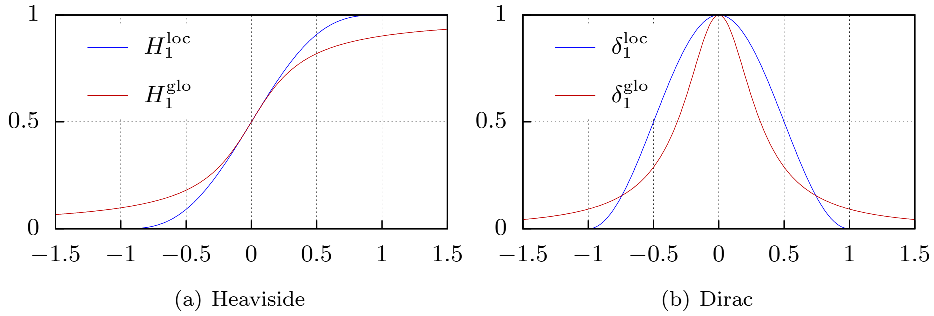
$$\boxed{\begin{aligned} H_\epsilon^{\mathrm{loc}}(x) &= \begin{cases} 0 & : x \leq -\epsilon \\ \frac{1}{2}\left(1 + \frac{x}{\epsilon} + \frac{1}{\pi} \cdot \sin \frac{\pi x}{\epsilon}\right) & : |x| < \epsilon \\ 1 & : x \geq \epsilon \end{cases} \\ \Rightarrow \quad \delta_\epsilon^{\mathrm{loc}}(x) &= \begin{cases} \frac{1}{2\epsilon}\left(1 + \cos \frac{\pi x}{\epsilon}\right) & : |x| < \epsilon \\ 0 & : |x| \geq \epsilon \end{cases} \end{aligned}}$$



with the properties

$$\begin{aligned} H_\epsilon^{\mathrm{loc}} &\in \mathcal{C}^2(\mathbf{R}) \\ \operatorname{supp} \delta_\epsilon^{\mathrm{loc}} &= (-\epsilon, \epsilon) \\ \delta_\epsilon^{\mathrm{loc}}(0) &= \frac{1}{\epsilon} \end{aligned}$$

in addition to (E.1)–(E.3).



(a) Heaviside (b) Dirac

Figure E.3: Mollified versions of the Heaviside step function and the Dirac distribution

Remark. Using the rather complicated definition of $H_\epsilon^{\mathrm{loc}}$ (instead of a plain shifted and scaled sin) ensures $\mathcal{C}^2$-smoothness at the definition interfaces.

List of (Co-)Authored Publications

Related to this Thesis

- C. Rohkohl, G. Lauritsch, A. Keil, and J. Hornegger. CAVAREV–an open platform for evaluating 3D and 4D cardiac vasculature reconstruction. *Physics in Med. and Biol.*, 55 (10):2905–2915, 2010. doi:10.1088/0031-9155/55/10/008.

- A. Keil, J. Vogel, G. Lauritsch, and N. Navab. Dynamic cone beam reconstruction using a new level set formulation. In *Proc. Int'l Conf. Med. Image Computing and Computer Assisted Intervention (MICCAI)*, vol. 5762 of *Lecture Notes in Computer Science*, pp. 389–397. Springer, Sept. 2009. doi:10.1007/978-3-642-04271-3_48.

- A. Keil, J. Vogel, G. Lauritsch, and N. Navab. Dynamic cone-beam reconstruction using a variational level set formulation. In *Proc. Int'l Meeting on Fully Three-Dim. Image Reconstr. in Radiol. and Nucl. Med. (Fully3D)*, pp. 319–322, Sept. 2009. `http://www.fully3d.org`.

- M. Blume, A. Martinez-Möller, A. Keil, N. Navab, and M. Rafecas. Joint reconstruction of image and motion in gated positron-emission-tomography. In *Proc. Int'l Meeting on Fully Three-Dim. Image Reconstr. in Radiol. and Nucl. Med. (Fully3D)*, pp. 118–121, Sept. 2009. `http://www.fully3d.org`.

- M. Blume, A. Keil, A. Martinez-Moeller, N. Navab, and M. Rafecas. Simultaneous reconstruction of image and motion in gated positron-emission-tomography. In *Proc. Nucl. Science Symp. and Med. Imag. Conf.*, pp. 2760–2762. IEEE, Oct. 2009. doi:10.1109/NSSMIC.2009.5401958.

- M. Blume, A. Keil, N. Navab, and M. Rafecas. Blind motion compensation for positron-emission-tomography. In *Proc. SPIE Med. Imaging*, vol. 7258 of *Proc. of the SPIE*. SPIE, Feb. 2009. doi:10.1117/12.811118.

Other Publications

- C. Wachinger, D. Mateus, A. Keil, and N. Navab. Manifold learning for patient position detection in MRI. In *Proc. IEEE Int'l Symp. on Biomed. Imag. (ISBI)*, pp. 1353–1356. IEEE, Apr. 2010. ISBN 978-1-4244-4126-6.

- C. Harrer, S. Kirchhoff, A. Keil, C. Kirchhoff, T. Mussack, A. Lienemann, M. Reiser, and N. Navab. Semi-automatic detection of Gd-DTPA-saline filled capsules for colonic transit time assessment in MRI. In J. M. Reinhardt and J. P. W. Pluim, editors, *Proc. SPIE Med. Imaging*, vol. 6914 of *Proc. of the SPIE*, p. 69143K, San Diego, CA, USA, 2008. SPIE. doi:10.1117/12.769867.

- L. König, M. Groher, A. Keil, C. Glaser, M. Reiser, and N. Navab. Semi-automatic segmentation fo the patellar cartilage in MRI. In A. Horsch, T. M. Deserno, and H. Handels, editors, *Proc. Bildverarbeitung für die Medizin (BVM)*, vol. 17 of *Informatik aktuell*, pp. 404–408. Springer, 2007. doi:10.1007/978-3-540-71091-2_81.
- A. Keil, C. Wachinger, G. Brinker, S. Thesen, and N. Navab. Patient position detection for SAR optimization in magnetic resonance imaging. In *Proc. Int'l Conf. Med. Image Computing and Computer Assisted Intervention (MICCAI)*, vol. 4191 of *Lecture Notes in Computer Science*, pp. 49–57. Springer, Oct. 2006. ISBN 3-540-44727-X. doi:10.1007/11866763_7.
- S. Vogt, A. Khamene, F. Sauer, A. Keil, and H. Niemann. A high performance AR system for medical applications. In *Proc. IEEE and ACM Int'l Symp. Mixed and Augmented Reality (ISMAR)*, pp. 270–271, Oct. 2003. doi:10.1109/ISMAR.2003.1240715. `http://portal.acm.org/citation.cfm?id=946248.946851`.

Bibliography

[1] S. Achenbach. Half-scan vs. multi-segment reconstruction for computed tomography coronary angiography. White paper, Siemens AG, Medical Solutions, CT, Siemensstr. 1, 91301 Forchheim, Germany, Dec. 2006. `http://www.medical.siemens.com/siemens/en_US/cs_cardio_FBAs/files/Case_Study/Achenbach_MultiSegment_Reconstruction.pdf`. Order No. A91CT-00352-12C1-7600.

[2] D. Adalsteinsson and J. A. Sethian. A fast level set method for propagating interfaces. *J. Computational Physics*, 118(2):269–277, 1995. doi:10.1006/jcph.1995.1098.

[3] D. Adalsteinsson and J. A. Sethian. The fast construction of extension velocities in level set methods. *J. Computational Physics*, 148(1):2–22, 1999. doi:10.1006/jcph.1998.6090.

[4] L. Ambrosio and H. M. Soner. Level set approach to mean curvature flow in arbitrary codimension. *J. of Diff. Geom.*, 43(4):693–737, 1996. `http://projecteuclid.org/euclid.jdg/1214458529`.

[5] A. H. Andersen and A. C. Kak. Simultaneous algebraic reconstruction technique (SART): A superior implementation of the ART algorithm. *Ultrasonic Imaging*, 6(1):81–94, 1984.

[6] G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*, vol. 147 of *Applied Mathematical Sciences*. Springer, 2nd edition, 2006. ISBN 978-0387-32200-1.

[7] S. Avidan and A. Shashua. Trajectory triangulation: 3D reconstruction of moving points from a monocular image sequence. *IEEE Trans. Patt. Anal. Machine Intell.*, 22(4):348–357, 2000. doi:10.1109/34.845377.

[8] B. G. Baumgart. *Geometric Modeling for Computer Vision.* Ph. D. thesis, Stanford Univ., Stanford, CA, USA, 1974.

[9] C. M. Bishop. *Pattern Recognition and Machine Learning.* Information Science and Statistics. Springer, 2nd edition, 2007. ISBN 978-0-387-31073-2.

[10] C. Blondel, G. Malandain, R. Vaillant, F. Devernay, È. Coste-Manière, and N. Ayache. 4-D tomographic representation of coronary arteries from one rotational X-ray sequence. In *Proc. Int'l Conf. Med. Image Computing and Computer Assisted Intervention (MICCAI)*, vol. 2878 of *Lecture Notes in Computer Science*, pp. 416–423. Springer, 2003. doi:10.1007/b93810.

[11] C. Blondel, R. Vaillant, G. Malandain, and N. Ayache. 3-D tomographic reconstruction of coronary arteries using a 4-D motion field. In *Proc. Int'l Meeting on Fully Three-Dim. Image Reconstr. in Radiol. and Nucl. Med. (Fully3D)*, 2003. `http://www.fully3d.org`.

[12] C. Blondel, R. Vaillant, G. Malandain, and N. Ayache. 3D tomographic reconstruction of coronary arteries using a precomputed 4D motion field. *Physics in Med. and Biol.*, 49 (11):2197–2208, 2004. doi:10.1088/0031-9155/49/11/006.

[13] C. Blondel, G. Malandain, R. Vaillant, and N. Ayache. Reconstruction of coronary arteries from a single rotational X-ray projection sequence. *IEEE Trans. Med. Imag.*, 25(5):653–663, 2006. doi:10.1109/TMI.2006.873224.

[14] M. Blume, A. Keil, A. Martinez-Moeller, N. Navab, and M. Rafecas. Simultaneous reconstruction of image and motion in gated positron-emission-tomography. In *Proc. Nucl. Science Symp. and Med. Imag. Conf.*, pp. 2760–2762. IEEE, Oct. 2009. doi:10.1109/NSSMIC.2009.5401958.

[15] M. Blume, A. Keil, N. Navab, and M. Rafecas. Blind motion compensation for positron-emission-tomography. In *Proc. SPIE Med. Imaging*, vol. 7258 of *Proc. of the SPIE*. SPIE, Feb. 2009. doi:10.1117/12.811118.

[16] M. Blume, A. Martinez-Möller, A. Keil, N. Navab, and M. Rafecas. Joint reconstruction of image and motion in gated positron-emission-tomography. In *Proc. Int'l Meeting on Fully Three-Dim. Image Reconstr. in Radiol. and Nucl. Med. (Fully3D)*, pp. 118–121, Sept. 2009. `http://www.fully3d.org`.

[17] T. Brox, A. Bruhn, and J. Weickert. Variational motion segmentation with level sets. In *Proc. Europ. Conf. Comp. Vis. (ECCV)*, vol. 3951 of *Lecture Notes in Computer Science*, pp. 471–483. Springer, May 2006. doi:10.1007/11744023_37.

[18] P. Buchard, L.-T. Cheng, B. Merriman, and S. Osher. Motion of curves in three spatial dimensions using a level set approach. *J. Computational Physics*, 170(2):720–741, 2001. doi:10.1006/jcph.2001.6758.

[19] T. M. Buzug. *Computed Tomography. From Photon Statistics to Modern Cone-Beam CT.* Springer, 2008. doi:10.1007/978-3-540-39408-2.

[20] V. Caselles, F. Catté, T. Coll, and F. Dibos. A geometric model for active contours in image processing. *Numerische Mathematik*, 66(1):1–31, 1993. doi:10.1007/BF01385685.

[21] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *Int'l J. Comp. Vis.*, 22(1):61–79, 1997. doi:10.1023/A:1007979827043.

[22] T. Chan and L. Vese. An active contour model without edges. In M. Nielsen, P. Johansen, O. F. Olsen, and J. Weickert, editors, *Proc. Scale-Space Theories in Comp. Vis.*, vol. 1682 of *Lecture Notes in Computer Science*, pp. 141–151. Springer, Sept. 1999. doi:10.1007/3-540-48236-9_13.

[23] T. F. Chan and L. A. Vese. Active contours without edges. *IEEE Trans. Image Process.*, 10(2):266–277, 2001. doi:10.1109/83.902291.

[24] T. F. Chan, S. Esedoglu, and M. Nikolova. Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM J. Appl. Math.*, 66(5):1632–1648, 2006. doi:10.1137/040615286.

[25] M. Chen, W. Lu, Q. Chen, K. J. Ruchala, and G. H. Olivera. A simple fixed-point approach to invert a deformation field. *Med. Phys.*, 35(1):81–88, 2008. doi:10.1118/1.2816107.

[26] K.-M. G. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette across time part I: Theory and algorithms. *Int'l J. Comp. Vis.*, 62(3):221–247, 2005. doi:10.1007/s11263-005-4881-5.

[27] K.-M. G. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette across time part II: Applications to human modeling and markerless motion tracking. *Int'l J. Comp. Vis.*, 63(3):225–245, 2005. doi:10.1007/s11263-005-6879-4.

[28] W. Chlewicki, C. Badea, and N. Pallikarakis. Cone based 3D reconstruction: A FDK-SART comparison for limited number of projections. In R. Magjarević, S. Tonković, V. Bilas, and I. Lacković, editors, *MEDICON 2001*, Proc. IFMBE, pp. 495–497. FER and CROMBES, University of Zagreb, Zagreb, Croatia, June 2001.

[29] A. M. Cormack. Representation of a function by its line integrals, with some radiological applications. *J. Appl. Physics*, 34(9):2722–2727, 1963. doi:10.1063/1.1729798.

[30] A. M. Cormack. Representation of a function by its line integrals, with some radiological applications. II. *J. Appl. Physics*, 35(10):2908–2913, 1964. doi:10.1063/1.1713127.

[31] D. Cremers. A variational framework for image segmentation combining motion estimation and shape regularization. In *Proc. IEEE Conf. Comp. Vis. and Patt. Recog. (CVPR)*, vol. 1, pp. 53–58, June 2003. doi:10.1109/CVPR.2003.1211337.

[32] D. Cremers. Dynamical statistical shape priors for level set-based tracking. *IEEE Trans. Patt. Anal. Machine Intell.*, 28(8):1262–1273, 2006. doi:10.1109/TPAMI.2006.161.

[33] D. Cremers. Nonlinear dynamical shape priors for level set segmentation. *J. Scientific Computing*, 35(2–3):132–143, 2008. doi:10.1007/s10915-008-9220-x.

[34] D. Cremers and C. Schnörr. Motion competition: Variational integration of motion segmentation and shape regularization. In *Pattern Recognition (Proc. DAGM)*, vol. 2449 of *Lecture Notes in Computer Science*, pp. 472–480. Springer, Sept. 2002. doi:10.1007/3-540-45783-6_57.

[35] D. Cremers and S. Soatto. Motion competition: A variational approach to piecewise parametric motion segmentation. *Int'l J. Comp. Vis.*, 62(3):249–265, 2005. doi:10.1007/s11263-005-4882-4.

[36] D. Cremers, M. Rousson, and R. Deriche. A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape. *Int'l J. Comp. Vis.*, 72(2):195–215, 2007. doi:10.1007/s11263-006-8711-1.

[37] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statistical Soc. Ser. B (Methodological)*, 39(1):1–38, 1977. `http://www.jstor.org/stable/2984875`.

[38] J. T. Dodge, Jr, B. G. Brown, E. L. Bolson, and H. T. Dodge. Lumen diameter of normal human coronary arteries. *Circulation*, 86:232–246, 1992. ISSN 0009-7322 (print), 1524-4539 (online).

[39] R. O. Duda and P. E. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Comm. of the ACM*, 15(11):11–15, 1972. doi:10.1145/361237.361242.

[40] L. C. Evans. *Partial Differential Equations.* Number 19 in Graduate Studies in Mathematics. AMS, 1st edition, 2008. ISBN 0-8218-0772-2.

[41] L. C. Evans and R. F. Gariepy. *Measure Theory and Fine Properties of Functions.* Studies in Advanced Mathematics. CRC Press, 1st edition, 1992. ISBN 0-8493-7157-0.

[42] R. Fahrig and D. W. Holdsworth. Three-dimensional computed tomographic reconstruction using a C-arm mounted XRII: Image-based correction of gantry motion nonidealities. *Med. Phys.*, 27(1):30–38, 2000. doi:10.1118/1.598854.

[43] R. Fahrig, M. Moreau, and D. W. Holdsworth. Three-dimensional computed tomographic reconstruction using a C-arm mounted XRII: Correction of image intensifier distortion. *Med. Phys.*, 24(7):1097–1106, 1997. doi:10.1118/1.598013.

[44] O. Faugeras and R. Keriven. Variational principles, surface evolution, PDE's, level set methods, and the stereo problem. *IEEE Trans. Patt. Anal. Machine Intell.*, 7(3):336–344, 1998. doi:10.1109/83.661183.

[45] R. Fedkiw. Homepage. `http://physbam.stanford.edu/~fedkiw/`.

[46] L. A. Feldkamp, L. C. Davis, and J. W. Kress. Practical cone-beam algorithm. *J. Opt. Soc. Am. A*, 1(6):612–619, 1984. doi:10.1364/JOSAA.1.000612.

[47] J.-S. Franco and E. Boyer. Fusion of multi-view silhouette cues using a space occupancy grid. In *Proc. Int'l Conf. Comp. Vis. (ICCV)*, vol. 2, pp. 1747–1753, 2005. doi:10.1109/ICCV.2005.105.

[48] A. F. Frangi, W. J. Niessen, K. L. Vincken, and M. A. Viergever. Multiscale vessel enhancement filtering. In *Proc. Int'l Conf. Med. Image Computing and Computer Assisted Intervention (MICCAI)*, vol. 1496 of *Lecture Notes in Computer Science*, pp. 130–137. Springer, Jan. 1998. doi:10.1007/BFb0056181.

[49] A. F. Frangi, W. J. Niessen, and M. A. Viergever. Three-dimensional modeling for functional analysis of cardiac images: A review. *IEEE Trans. Med. Imag.*, 20(1):2–25, 2001. doi:10.1109/42.906421.

[50] S. I. Frolov. Continuous products and random walks. *J. Math. Sciences*, 88(6):884–893, 1998. doi:10.1007/BF02365376. Translation from "Statisticheski Metody Otsenivaniya i Proverki Gipotez", pp. 237–252, Perm, Russia, 1995.

[51] J. A. Garcia, P. Agostoni, N. E. Green, J. T. Maddux, J. S.-Y. Chen, J. C. Messenger, I. P. Casserly, A. Hansgen, O. Wink, B. Movassaghi, B. M. Groves, P. van den Heuvel, S. Verheye, G. van Langenhove, P. Vermeersch, F. van den Branden, Y. Yeghiazarians, A. D. Michaels, and J. D. Carroll. Rotational vs. standard coronary angiography: An image content analysis. *J. Catheterization and Cardiovasc. Interv.*, 73(6):753–671, 2009. doi:10.1002/ccd.21918.

[52] R. J. Gardner. *Geometric Tomography*, vol. 58 of *Encyclopedia of Mathematics and its Applications*. Cambridge Univ. Press, 2 edition, 2006. ISBN 0521684935.

[53] I. V. Girsanov. *Lectures on Mathematical Theory of Extremum Problems*. Number 67 in Lecture Notes in Economics and Mathematical Systems. Springer, 1972. ISBN 3-540-05857-5.

[54] J. Gomes and O. Faugeras. Reconciling distance functions and level sets. In M. Nielsen, P. Johansen, O. F. Olsen, and J. Weickert, editors, *Proc. Scale-Space Theories in Comp. Vis.*, vol. 1682 of *Lecture Notes in Computer Science*, pp. 70–81. Springer, Sept. 1999. doi:10.1007/3-540-48236-9_7.

[55] J. Gomes and O. Faugeras. Reconciling distance functions and level sets. *J. Visual Communication and Image Representation*, 11(2):209–223, 2000. doi:10.1006/jvci.1999.0439.

[56] A. Gooya, H. Liao, K. Matsumiya, K. Masamune, Y. Masutani, and T. Dohi. A variational method for geometric regularization of vascular segmentation in medical images. *IEEE Trans. Image Process.*, 17(8):1295–1312, 2008. doi:10.1109/TIP.2008.925378.

[57] R. Gordon, R. Bender, and G. T. Herman. Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography. *J. Theoretical Biol.*, 29 (3):471–481, 1970. doi:10.1016/0022-5193(70)90109-8.

[58] M. Grass, R. Manzke, T. Nielsen, P. Koken, R. Proksa, M. Natanzon, and G. Shechter. Helical cardiac cone beam reconstruction using retrospective ECG gating. *Physics in Med. and Biol.*, 48(18):3069–3084, 2003. doi:10.1088/0031-9155/48/18/308.

[59] N. E. Green, J. S.-Y. Chen, A. R. Hansgen, J. C. Messenger, B. M. Groves, and J. D. Carroll. Angiographic views used for percutaneous coronary interventions: A three-dimensional analysis of physician-determined vs. computer-generated views. *J. Catheterization and Cardiovasc. Interv.*, 64(4):451–459, 2005. doi:10.1002/ccd.20331.

[60] E. Hansis, D. Schäfer, M. Grass, and O. Dössel. An iterative method for the reconstruction of the coronary arteries from rotational X-ray angiography. In *Proc. SPIE Med. Imaging*, vol. 6510 of *Proc. of the SPIE*, pp. 651026–1–651026–10. SPIE, Feb. 2007. doi:10.1117/12.709319.

[61] E. Hansis, D. Schäfer, O. Dössel, and M. Grass. Evaluation of iterative sparse object reconstruction from few projections for 3-D rotational coronary angiography. *IEEE Trans. Med. Imag.*, 27(11):1548–1555, 2008. doi:10.1109/TMI.2008.2006514.

[62] E. Hansis, D. Schäfer, O. Dössel, and M. Grass. Projection-based motion compensation for gated coronary artery reconstruction from rotational x-ray angiograms. *Physics in Med. and Biol.*, 53:3807–3820, 2008. doi:10.1088/0031-9155/53/14/007.

[63] E. Hansis, H. Schomberg, K. Erhard, O. Dössel, and M. Grass. Four-dimensional cardiac reconstruction from rotational X-ray sequences - first results for 4D coronary angiography. In *Proc. SPIE Med. Imaging*, vol. 7258 of *Proc. of the SPIE*, pp. 72580B–1–72580B–11. SPIE, Feb. 2009. doi:10.1117/12.811104.

[64] R. Hartley and A. Zisserman. *Multiple View Geometry.* Cambridge Univ. Press, 2 edition, 2003. ISBN 0-521-54051-8.

[65] G. T. Herman. *Fundamentals of Computerized Tomography. Image Reconstruction from Projections.* Advances in Pattern Recognition. Springer, 2nd edition, 2009. ISBN 978-1-85233-617-2 (Print) 978-1-84628-723-7 (Online). doi:10.1007/978-1-84628-723-7.

[66] G. T. Herman and A. Kuba. *Advances in Discrete Tomography and its Applications.* Applied and Numerical Harmonic Analysis. Birkhäuser, 2007. ISBN 0817636145.

[67] H. P. Ho, Y. Chen, H. Liu, and P. Shi. Level set active contours on unstructured point cloud. In *Proc. IEEE Conf. Comp. Vis. and Patt. Recog. (CVPR)*, vol. 2, pp. 655–662. IEEE, June 2005. doi:10.1109/CVPR.2005.211.

[68] P. V. C. Hough. Method and means for recognizing complex patterns. U.S. Patent 3,069,654, Dec. 18 1962.

[69] G. N. Hounsfield. Computerized transverse axial scanning (tomography): Part I. description of system. *British J. Radiol.*, 46(552):1016–1022, 1973.

[70] X. Huang and D. N. Metaxas. Metamorphs: Deformable shape and appearance models. *IEEE Trans. Patt. Anal. Machine Intell.*, 30(8):1444–1459, 2008. doi:10.1109/TPAMI.2007.70795.

[71] X. Huang, D. Metaxas, and T. Chen. Metamorphs: Deformable shape and texture models. In *Proc. IEEE Conf. Comp. Vis. and Patt. Recog. (CVPR)*, vol. 1, pp. I–496–I–503. IEEE, June 2004. doi:10.1109/CVPR.2004.1315072.

[72] L. Husmann, S. Leschka, L. Desbiolles, T. Schepis, O. Gaemperli, B. Seifert, P. Cattin, T. Frauenfelder, T. G. Flohr, B. Marincek, P. A. Kaufmann, and H. Alkadhi. Coronary artery motion and cardiac phases: Dependency on heart rate – implications for CT image reconstruction. *Radiology*, 245(2):567–576, 2007. doi:10.1148/radiol.2451061791.

[73] A. A. Isola, A. Ziegler, T. Koehler, W. J. Niessen, and M. Grass. Motion-compensated iterative cone-beam ct image reconstruction with adapted blobs as basis functions. *Physics in Med. and Biol.*, 53:6777–6797, 2008. doi:10.1088/0031-9155/53/23/009.

[74] A. K. Jain, T. Mustafa, Y. Zhou, C. Burdette, G. S. Chirikjian, and G. Fichtinger. FTRAC – a robust fluoroscope tracking fiducial. *Med. Phys.*, 32(10):3185–3198, 2005. doi:10.1118/1.2047782.

[75] U. Jandt, D. Schäfer, M. Grass, and V. Rasche. Automatic generation of time resolved 4D motion vector fields of coronary arteries. In *Proc. Int'l Meeting on Fully Three-Dim. Image Reconstr. in Radiol. and Nucl. Med. (Fully3D)*, pp. 249–252, July 2007. `http://www.fully3d.org`.

[76] U. Jandt, D. Schäfer, V. Rasche, and M. Grass. Automatic generation of 3D coronary artery centerlines. In *Proc. SPIE Med. Imaging*, vol. 6510 of *Proc. of the SPIE*, pp. 65104Y–1–65104Y–8. SPIE, Feb. 2007. doi:10.1117/12.708941.

[77] U. Jandt, D. Schäfer, M. Grass, and V. Rasche. Automatic generation of 3D coronary artery centerlines using rotational X-ray angiography. *Med. Img. Anal.*, 13(6):846–858, 2009. doi:10.1016/j.media.2009.07.010.

[78] H. Jin, A. J. Yezzi, Y.-H. Tsai, L.-T. Cheng, and S. Soatto. Estimation of 3D surface shape and smooth radiance from 2D images: A level set approach. *J. Scientific Computing*, 19 (1–3):267–292, 2003. doi:10.1023/A:1025308109816.

[79] H. Jin, D. Cremers, A. J. Yezzi, and S. Soatto. Shedding light on stereoscopic segmentation. In *Proc. IEEE Conf. Comp. Vis. and Patt. Recog. (CVPR)*, vol. 1, pp. 36–42, June 2004. doi:10.1109/CVPR.2004.1315011.

[80] M. Kachelriess and W. A. Kalender. Electrocardiogram-correlated image reconstruction from subsecond spiral computed tomography scans of the heart. *Med. Phys.*, 25(12): 2417–2431, 1998. doi:10.1118/1.598453.

[81] M. Kachelriess, S. Ulzheimer, and W. A. Kalender. ECG-correlated imaging of the heart with subsecond multislice spiral CT. *IEEE Trans. Med. Imag.*, 19(9):888–901, 2000. doi:10.1109/42.887837.

[82] S. Kaczmarz. Przybliżone rozwiązywanie układów równań liniowych. – Angenäherte Auflösung von Systemen linearer Gleichungen. *Bulletin International de l'Académie Polonaise des Sciences et des Lettres. Classe des Sciences Mathématiques et Naturelles. Série A, Sciences Mathématiques*, pp. 355–357, 1937. Attention: This is commonly cited with the *erroneous* page range 335–357.

[83] A. C. Kak and M. Slaney. *Principles of Computerized Tomographic Imaging.* Number 33 in Classics in Applied Mathematics. SIAM, 2001. ISBN 0-89871-494-X.

[84] J. Y. Kaminski and A. Shashua. Multiple view geometry of general algebraic curves. *Int'l J. Comp. Vis.*, 56(3):195–219, 2004. doi:10.1023/B:VISI.0000011204.89453.4d.

[85] J. Y. Kaminski and M. Teicher. A general framework for trajectory triangulation. *J. Math. Imaging and Vis.*, 21(1–2):27–41, 2004. doi:10.1023/B:JMIV.0000026555.79056.b8.

[86] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *Int'l J. Comp. Vis.*, 1(4):321–331, 1988. doi:10.1007/BF00133570.

[87] A. Keil, J. Vogel, G. Lauritsch, and N. Navab. Dynamic cone-beam reconstruction using a variational level set formulation. In *Proc. Int'l Meeting on Fully Three-Dim. Image Reconstr. in Radiol. and Nucl. Med. (Fully3D)*, pp. 319–322, Sept. 2009. `http://www.fully3d.org`.

[88] A. Keil, J. Vogel, G. Lauritsch, and N. Navab. Dynamic cone beam reconstruction using a new level set formulation. In *Proc. Int'l Conf. Med. Image Computing and Computer Assisted Intervention (MICCAI)*, vol. 5762 of *Lecture Notes in Computer Science*, pp. 389–397. Springer, Sept. 2009. doi:10.1007/978-3-642-04271-3_48.

[89] C. Kirbas and F. Quek. A review of vessel extraction techniques and algorithms. *ACM Computing Surveys*, 36(2):81–121, 2004. doi:10.1145/1031120.1031121.

[90] M. Klodt, T. Schoenemann, K. Kolev, M. Schikora, and D. Cremers. An experimental comparison of discrete and continuous shape optimization methods. In *Proc. Europ. Conf. Comp. Vis. (ECCV)*, vol. 5302 of *Lecture Notes in Computer Science*, pp. 332–345. Springer, 2008. doi:10.1007/978-3-540-88682-2_26.

[91] L. P. Kobbelt, M. Botsch, U. Schwanecke, and H.-P. Seidel. Feature sensitive surface extraction from volume data. In *Proc. SIGGRAPH*, pp. 57–66. ACM, 2001. ISBN 1-58113-374-X. doi:10.1145/383259.383265.

[92] K. Kolev and D. Cremers. Integration of multiview stereo and silhouettes via convex functionals on convex domains. In *Proc. Europ. Conf. Comp. Vis. (ECCV)*, vol. 5302 of *Lecture Notes in Computer Science*, pp. 752–765. Springer, 2008. doi:10.1007/978-3-540-88682-2_57.

[93] K. Kolev, T. Brox, and D. Cremers. Robust variational segmentation of 3D objects from multiple views. In *Pattern Recognition (Proc. DAGM)*, vol. 4174 of *Lecture Notes in Computer Science*, pp. 688–697. Springer, 2006. doi:10.1007/11861898_69.

[94] K. Kolev, M. Klodt, T. Brox, S. Esedoglu, and D. Cremers. Continuous global optimization in multiview 3D reconstruction. In *Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, vol. 4679 of *Lecture Notes in Computer Science*, pp. 441–452. Springer, Aug. 2007. doi:10.1007/978-3-540-74198-5_34.

[95] K. Kolev, M. Klodt, T. Brox, and D. Cremers. Continuous global optimization in multiview 3D reconstruction. *Int'l J. Comp. Vis.*, 84:80–96, 2009. doi:10.1007/s11263-009-0233-1.

[96] T. M. Koller, G. Gerig, G. Székely, and D. Dettwiler. Multiscale detection of curvilinear structures in 2-D and 3-D image data. In *Proc. Int'l Conf. Comp. Vis. (ICCV)*, pp. 864–869, June 1995. doi:10.1109/ICCV.1995.466846.

[97] K. Krissian, G. Malandain, N. Ayache, R. Vaillant, and Y. Trousset. Model-based detection of tubular structures in 3D images. *J. Comp. Vis. and Img. Understanding*, 80(2):130–171, 2000. doi:10.1006/cviu.2000.0866.

[98] G. Lauritsch, J. Boese, L. Wigström, H. Kenneth, and R. Fahrig. Towards cardiac C-arm computed tomography. *IEEE Trans. Med. Imag.*, 25(7):922–934, 2006. doi:10.1109/TMI.2006.876166.

[99] G. Lauritsch, J. Boese, L. Wigström, M. Prümmer, and R. Fahrig. Temporal resolution in cardiac C-arm CT in the presence of variable heart rate. In *Proc. Int'l Meeting on Fully Three-Dim. Image Reconstr. in Radiol. and Nucl. Med. (Fully3D)*, pp. 358–361, July 2007. `http://www.fully3d.org`.

[100] M. E. Leventon, W. E. L. Grimson, and O. Faugeras. Statistical shape influence in geodesic active contours. In *Proc. IEEE Conf. Comp. Vis. and Patt. Recog. (CVPR)*, vol. 1, pp. 316–323, June 2000. doi:10.1109/CVPR.2000.855835.

[101] C. Li, C. Xu, C. Gui, and M. D. Fox. Level set evolution without re-initialization: A new variational formulation. In *Proc. IEEE Conf. Comp. Vis. and Patt. Recog. (CVPR)*, vol. 1, pp. 430–436. IEEE, 2005. doi:10.1109/CVPR.2005.213.

[102] C. Li, C. Xu, K. M. Konwar, and M. D. Fox. Fast distance preserving level set evolution for medical image segmentation. In *Int'l. Conf. Control, Automation, Robotics and Vision*, pp. 1–7. Springer, Dec. 2006. doi:10.1109/ICARCV.2006.345357.

[103] D. Lloyd-Jones, R. Adams, M. Carnethon, G. De Simone, T. B. Ferguson, K. Flegal, E. Ford, K. Furie, A. Go, K. Greenlund, N. Haase, S. Hailpern, M. Ho, V. Howard, B. Kissela, S. Kittner, D. Lackland, L. Lisabeth, A. Marelli, M. McDermott, J. Meigs, D. Mozaffarian, G. Nichol, C. O'Donnell, V. Roger, W. Rosamond, R. Sacco, P. Sorlie, R. Stafford, J. Steinberger, T. Thom, S. Wasserthiel-Smoller, N. Wong, J. Wylie-Rosett, and Y. Hong. Heart disease and stroke statistics 2009 update: A report from the American Heart Association Statistics Committee and Stroke Statistics Subcommittee. *Circulation*, 119(3):e21–e181, 2009. doi:10.1161/CIRCULATIONAHA.108.191261.

[104] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *Proc. SIGGRAPH*, vol. 21, pp. 163–169. ACM, 1987. doi:10.1145/37401.37422.

[105] L. M. Lorigo, O. Faugeras, W. E. L. Grimson, R. Keriven, R. Kikinis, and C.-F. Westin. Co-dimension 2 geodesic active contours for MRA segmentation. In A. Kuba, M. Šámal, and A. Todd-Pokropek, editors, *Int'l Conf. Information Proc. Med. Imag. (IPMI)*, vol. 1613 of *Lecture Notes in Computer Science*, pp. 126–139, Visegrád, Hungary, June 1999. Springer. doi:10.1007/3-540-48714-X_10.

[106] J. T. Maddux, J. S.-Y. Chen, B. M. Groves, J. C. Messenger, O. Wink, and J. D. Carroll. Rotational angiography and 3D coronary modeling: revolutions in the cardiac cath lab. *MedicaMundi*, 47(2):8–14, Aug. 2003. `http://www.medical.philips.com/pwc_hc/in_en/about/assets/Docs/medicamundi/mm_vol47_no2/08-14_Maddux_38487.pdf`.

[107] J. T. Maddux, O. Wink, J. C. Messenger, B. M. Groves, R. Liao, J. Strzelczyk, S.-Y. Chen, and J. D. Carroll. Randomized study of the safety and clinical utility of rotational angiography versus standard angiography in the diagnosis of coronary artery disease. *Cath. and Cardiovasc. Interv.*, 62(2):167–174, 2004. doi:10.1002/ccd.20036.

[108] R. Malladi, J. A. Sethian, and B. C. Vemuri. Shape modeling with front propagation: A level set approach. *IEEE Trans. Patt. Anal. Machine Intell.*, 17(2):158–175, 1995. doi:10.1109/34.368173.

[109] R. Malladi, R. Kimmel, D. Adalsteinsson, G. Sapiro, V. Caselles, and J. A. Sethian. A geometric approach to segmentation and analysis of 3D medical images. In *Proc. Workshop on Math. Methods in Biomed. Image Anal.*, pp. 244–252, June 1996. doi:10.1109/MMBIA.1996.534076.

[110] B. S. Morse, W. Liu, T. S. Yoo, and K. Subramanian. Active contours using a constraint-based implicit representation. In *Proc. IEEE Conf. Comp. Vis. and Patt. Recog. (CVPR)*, vol. 1, pp. 285–292. IEEE, June 2005. doi:10.1109/CVPR.2005.59.

[111] B. Movassaghi, V. Rasche, M. Grass, M. A. Viergever, and W. J. Niessen. A quantitative analysis of 3-D coronary modeling from two or more projection images. *IEEE Trans. Med. Imag.*, 23(12):1517–1531, 2004. doi:10.1109/TMI.2004.837340.

[112] B. Movassaghi, V. Rasche, M. A. Viergever, and W. J. Niessen. 3D coronary reconstruction from calibrated motion-compensated 2D projections based on semi-automated feature point detection. In *Proc. SPIE Med. Imaging*, vol. 5370 of *Proc. of the SPIE*, pp. 1943–1950. SPIE, 2004. doi:10.1117/12.535271.

[113] B. Movassaghi, M. Grass, D. Schaefer, V. Rasche, O. Wink, G. Schoonenberg, J. Y. Chen, J. A. Garcia, B. M. Groves, J. C. Messenger, and J. D. Carroll. 4D coronary artery reconstruction based on retrospectively gated rotational angiography: first in-human results. In *Proc. SPIE Med. Imaging*, vol. 6509 of *Proc. of the SPIE*, pp. 65090P–1–65090P–6. SPIE, 2007. doi:10.1117/12.706435.

[114] K. Mueller. *Fast and Accurate Three-Dimensional Reconstruction from Cone-Beam Projection Data Using Algebraic Methods.* Ph. D. thesis, Ohio State University, 1998.

[115] F. Natterer. *The Mathematics of Computerized Tomography.* John Wiley & Sons, 1986. ISBN 0 471 90959 9.

[116] F. Natterer and F. Wübbeling. *Mathematical Methods in Image Reconstruction.* Monographs on Mathematical Modeling and Computation. SIAM, 2001. ISBN 0898714729.

[117] N. Navab, A. R. Bani-Hashemi, M. M. Mitschke, D. W. Holdsworth, R. Fahrig, A. J. Fox, and R. Graumann. Dynamic geometrical calibration for 3-D cerebral angiography. In *Proc. SPIE Med. Imaging*, vol. 2708 of *Proc. of the SPIE*, pp. 361–370. SPIE, Feb. 1996. doi:10.1117/12.237798.

[118] N. Navab, A. Bani-Hashemi, M. S. Nadar, K. Wiesent, P. Durlak, T. Brunner, K. Barth, and R. Graumann. 3D reconstruction from projection matrices in a C-arm based 3D-angiography system. In *Proc. Int'l Conf. Med. Image Computing and Computer Assisted Intervention (MICCAI)*, vol. 1496 of *Lecture Notes in Computer Science*, pp. 119–129. Springer, Jan. 1998. doi:10.1007/BFb0056181.

[119] openmpweb. OpenMP website. `http://www.openmp.org`.

[120] openmpwiki. Wikipedia article on OpenMP. `http://en.wikipedia.org/wiki/OpenMP`.

[121] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces.* Number 153 in Applied Mathematical Sciences. Springer, 2003. ISBN 0-387-95482-1.

[122] S. Osher and R. P. Fedkiw. Level set methods: An overview and some recent results. *J. Computational Physics*, 169(2):463–502, 2001. doi:10.1006/jcph.2000.6636.

[123] S. Osher and N. Paragios, editors. *Geometric Level Set Methods in Imaging, Vision, and Graphics.* Springer, 2003. ISBN 0-387-95488-0.

[124] S. Osher and J. A. Sethian. Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J. Computational Physics*, 79(1): 12–49, 1988. doi:10.1016/0021-9991(88)90002-2.

[125] X. Pan, E. Y. Sidky, and M. Vannier. Why do commercial CT scanners still employ traditional, filtered back-projection for image reconstruction? *J. Inverse Problems*, 25 (12):123009.1–123009.36, 2009. doi:10.1088/0266-5611/25/12/123009.

[126] D. Peng, B. Merriman, S. Osher, H. Zhao, and M. Kang. A PDE-based fast local level set method. *J. Computational Physics*, 155(2):410–438, 1999. doi:10.1006/jcph.1999.6345.

[127] J.-P. Pons, R. Keriven, and O. Faugeras. Modelling dynamic scenes by registering multi-view image sequences. In *Proc. IEEE Conf. Comp. Vis. and Patt. Recog. (CVPR)*, vol. 2, pp. 822–827, June 2005. doi:10.1109/CVPR.2005.227.

[128] J.-P. Pons, R. Keriven, and O. Faugeras. Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. *Int'l J. Comp. Vis.*, 72(2):179–193, 2007. doi:10.1007/s11263-006-8671-5.

[129] M. Prümmer, L. Wigstroem, J. Hornegger, J. Boese, G. Lauritsch, N. Strobel, and R. Fahrig. Cardiac C-arm CT: Efficient motion correction for 4D-FBP. In *Proc. Nucl. Science Symp. and Med. Imag. Conf.*, pp. 2620–2628. IEEE, 2006. doi:10.1109/NSSMIC.2006.354444.

[130] M. Prümmer, R. Fahrig, L. Wigström, J. Boese, and G. Lauritsch. Cardiac C-arm CT: 4D non-model based heart motion estimation and its application. In *Proc. SPIE Med. Imaging*, vol. 6510 of *Proc. of the SPIE*, pp. 651015–1–651015–12. SPIE, 2007. doi:10.1117/12.708151.

[131] M. Prümmer, J. Hornegger, G. Lauritsch, L. Wigström, E. Girard-Hughes, and R. Fahrig. Cardiac C-arm CT: a unified framework for motion estimation and dynamic CT. *IEEE Trans. Med. Imag.*, 28(11):1836–1849, 2009. doi:10.1109/TMI.2009.2025499.

[132] J. Radon. Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten. *Berichte über die Verhandlungen der Sächsischen Akademie der Wissenschaften zu Leipzig. Mathematisch-Physische Klasse*, 69:262–277, 1917. See [133] for an English translation.

[133] J. Radon. On the determination of functions from their integral values along certain manifolds. *IEEE Trans. Med. Imag.*, 5(4):170–176, 1986. doi:10.1109/TMI.1986.4307775. Translation by P.C. Parks from the original article [132].

[134] C. Rohkohl, G. Lauritsch, A. Nöttling, M. Prümmer, and J. Hornegger. C-arm CT: Reconstruction of dynamic high contrast objects applied to the coronary sinus. In *Proc. Nucl. Science Symp. and Med. Imag. Conf.*, pp. 5113–5120. IEEE, Oct. 2008. doi:10.1109/NSSMIC.2008.4774387.

[135] C. Rohkohl, M. Prümmer, G. Lauritsch, and J. Hornegger. Cardiac C-arm CT: image-based gating. In *Proc. SPIE Med. Imaging*, vol. 6913 of *Proc. of the SPIE*, pp. 69131G–1–69131G–12. SPIE, 2008. doi:10.1117/12.770124.

[136] C. Rohkohl, G. Lauritsch, M. Prümmer, and J. Hornegger. Interventional 4-D motion estimation and reconstruction of cardiac vasculature without motion periodicity assumption. In *Proc. Int'l Conf. Med. Image Computing and Computer Assisted Intervention (MICCAI)*, vol. 5761 of *Lecture Notes in Computer Science*, pp. 132–139. Springer, Sept. 2009. doi:10.1007/978-3-642-04268-3_17.

[137] C. Rohkohl, G. Laurtisch, M. Prümmer, J. Boese, and J. Hornegger. Towards 4-D cardiac reconstruction without ECG and motion periodicity using C-arm CT. In *Proc. Int'l Meeting on Fully Three-Dim. Image Reconstr. in Radiol. and Nucl. Med. (Fully3D)*, pp. 323–326, Sept. 2009. `http://www.fully3d.org`.

[138] C. Rohkohl, G. Lauritsch, A. Keil, and J. Hornegger. CAVAREV–an open platform for evaluating 3D and 4D cardiac vasculature reconstruction. *Physics in Med. and Biol.*, 55 (10):2905–2915, 2010. doi:10.1088/0031-9155/55/10/008.

[139] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1–4):259–268, 1992. doi:10.1016/0167-2789(92)90242-F.

[140] G. Sapiro. *Geometric Partial Differential Equations and Image Analysis.* Cambridge Univ. Press, 2001. ISBN 0-521-79075-1.

[141] D. Schäfer, J. Borgert, V. Rasche, and M. Grass. Motion-compensated and gated cone beam filtered back-projection for 3-D rotational X-ray angiography. *IEEE Trans. Med. Imag.*, 25(7):898–906, 2006. doi:10.1109/TMI.2006.876147.

[142] D. Schäfer, U. Jandt, J. D. Carroll, and M. Grass. Motion compensated reconstruction for rotational X-ray angiography using 4D coronary centerline models. In *Proc. Int'l Meeting on Fully Three-Dim. Image Reconstr. in Radiol. and Nucl. Med. (Fully3D)*, pp. 245–248, July 2007. `http://www.fully3d.org`.

[143] T. Schoenemann and D. Cremers. Near real-time motion segmentation using graph cuts. In *Pattern Recognition (Proc. DAGM)*, vol. 4174 of *Lecture Notes in Computer Science*, pp. 455–464. Springer, Sept. 2006. doi:10.1007/11861898_46.

[144] H. Schomberg. Time-resolved cardiac cone beam CT. In *Proc. Int'l Meeting on Fully Three-Dim. Image Reconstr. in Radiol. and Nucl. Med. (Fully3D)*, pp. 362–365, July 2007. `http://www.fully3d.org`.

[145] W. P. Segars, M. Mahesh, T. Beck, E. Frey, and B. Tsui. Realistic CT simulation using the 4D XCAT phantom. *Med. Phys.*, 35(8):3800–3808, 2008. doi:10.1118/1.2955743.

[146] J. A. Sethian. Homepage. `http://math.berkeley.edu/~sethian/`.

[147] J. A. Sethian. Numerical algorithms for propagating interfaces: Hamilton-Jacobi equations and conservation laws. *J. of Diff. Geom.*, 31(1):131–161, 1990. `http://projecteuclid.org/euclid.jdg/1214444092`.

[148] J. A. Sethian. Theory, algorithms, and applications of level set methods for propagating interfaces. *Acta Numerica*, pp. 309–395, 1996.

[149] J. A. Sethian. A fast marching level set method for monotonically advancing fronts. *Proc. Nat. Acad. Sci.*, 93(4):1591–1595, 1996. doi:10.1073/pnas.93.4.1591.

[150] J. A. Sethian. *Level Set Methods and Fast Marching Methods. Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science.* Number 3 in Cambridge Monographs on Applied and Computational Mathematics. Cambridge Univ. Press, 2nd edition, 1999. ISBN 0-521-64557-3.

[151] J. A. Sethian. Evolution, implementation, and application of level set and fast marching methods for advancing fronts. *J. Computational Physics*, 169(2):503–555, 2001. doi:10.1006/jcph.2000.6657.

[152] J. A. Sethian and J. Straint. Crystal growth and dendritic solidification. *J. Computational Physics*, 98(2):231–253, 1992. doi:10.1016/0021-9991(92)90140-T.

[153] L. A. Shepp and B. F. Logan. Reconstructing interior head tissue from X-ray transmissions. *IEEE Trans. Nucl. Science*, 21(1):228–236, Feb. 1974. doi:10.1109/TNS.1974.4327466. See [154] for a corrected and expanded version.

[154] L. A. Shepp and B. F. Logan. The Fourier reconstruction of a head section. *IEEE Trans. Nucl. Science*, 21(3):21–46, June 1974. Corrected and expanded version of [153].

[155] L. A. Shepp and Y. Vardi. Maximum likelihood reconstruction for emission tomography. *IEEE Trans. Med. Imag.*, 1(2):113–122, 1982. doi:10.1109/TMI.1982.4307558.

[156] E. Y. Sidky and X. Pan. Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization. *Physics in Med. and Biol.*, 53(17):4777–4807, 2008. doi:10.1088/0031-9155/53/17/021.

[157] E. Y. Sidky, C.-M. Kao, and X. Pan. Accurate image reconstruction from few-views and limited-angle data in divergent-beam CT. *J. X-Ray Sci. Tech.*, 14(2):119–139, 2006. `http://arxiv.org/abs/0904.4495v1`.

[158] E. Y. Sidky, R. Chartrand, and X. Pan. Image reconstruction from few views by non-convex optimization. In *Proc. Nucl. Science Symp. and Med. Imag. Conf.*, vol. 5, pp. 3526–3530. IEEE, 2007. doi:10.1109/NSSMIC.2007.4436889.

[159] G. Slabaugh, Q. Dinh, and G. Unal. A variational approach to the evolution of radial basis functions for image segmentation. In *Proc. IEEE Conf. Comp. Vis. and Patt. Recog. (CVPR)*, pp. 1–8. IEEE, June 2007. doi:10.1109/CVPR.2007.383013.

[160] J. P. Stonestrom, R. E. Alvarez, and A. Macovski. A framework for spectral artifact corrections in x-ray ct. *IEEE Trans. Biomed. Eng.*, BME-28(2):128–141, 1981. doi:10.1109/TBME.1981.324786.

[161] T. Strohmer and R. Vershynin. A randomized Kaczmarz algorithm for linear systems with exponential convergence. *J. Fourier Analysis and Appl.*, 15(2):262–278, 2009. doi:10.1007/s00041-008-9030-4.

[162] M. Sussman and E. Fatemi. An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. *SIAM J. Sci. Comp.*, 20(4):1165–1191, 1999. doi:10.1137/S1064827596298245.

[163] M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *J. Computational Physics*, 114(1):146–159, 1994. doi:10.1006/jcph.1994.1155.

[164] R. Szeliski. Rapid octree construction from image sequences. *J. Comp. Vis., Graphics and Img. Proc.: Img. Understanding*, 58(1):23–32, 1993. doi:10.1006/ciun.1993.1029.

[165] H. Turbell. *Cone-Beam Reconstruction Using Filtered Backprojection.* Ph. D. thesis, Linköpings Universitet, Linköping, Sweden, Feb. 2001.

[166] L. A. Vese and T. F. Chan. A multiphase level set framework for image segmentation using the Mumford and Shah model. *Int'l J. Comp. Vis.*, 50(3):271–293, 2002. doi:10.1023/A:1020874308076.

[167] J. Vogel. Symbolic 3D+t reconstruction from cone-beam projections. Diploma thesis, Technische Universität München, Fakultät für Informatik, München, Germany, Sept. 2009.

[168] R. T. Whitaker. A level-set approach to 3D reconstruction from range data. *Int'l J. Comp. Vis.*, 29(3):203–231, Sept. 1998. doi:10.1023/A:1008036829907.

[169] WHO. The atlas of heart disease and stroke. `http://www.who.int/cardiovascular_diseases/resources/atlas/`.

[170] K. Wiesent, K. Barth, N. Navab, P. Durlak, T. Brunner, O. Schuetz, and W. Seissler. Enhanced 3-D-reconstruction algorithm for c-arm systems suitable for interventional procedures. *IEEE Trans. Med. Imag.*, 19(5):391–403, 2000. doi:10.1109/42.870250.

[171] C. Xu and J. L. Prince. Snakes, shapes, and gradient vector flow. *IEEE Trans. Image Process.*, 7(3):359–369, 1998. doi:10.1109/83.661186.

[172] C. Xu, A. Yezzi, Jr, and J. L. Prince. On the relationship between parametric and geometric active contours. In *Proc. Asilomar Conf. Signals, Systems and Computers*, pp. 483–489, Oct. 2000. doi:10.1109/ACSSC.2000.911003.

[173] C. Xu, A. Yezzi, Jr, and J. L. Prince. A summary of geometric level-set analogues for a general class of parametric active contour and surface models. In *Proc. IEEE Workshop Variational Level Set Methods in Comp. Vis. (VLSM)*, pp. 104–111. IEEE, July 2001. doi:10.1109/VLSM.2001.938888.

[174] A. Yezzi and S. Soatto. Stereoscopic segmentation. In *Proc. Int'l Conf. Comp. Vis. (ICCV)*, vol. 1, pp. 59–66, July 2001. doi:10.1109/ICCV.2001.937499.

[175] A. Yezzi and S. Soatto. Stereoscopic segmentation. *Int'l J. Comp. Vis.*, 53(1):31–43, 2003. doi:10.1023/A:1023079624234.

[176] A. Yezzi, L. Zöllei, and T. Kapur. A variational framework for joint segmentation and registration. In *Proc. Workshop on Math. Methods in Biomed. Image Anal.*, pp. 44–51, Dec. 2001. doi:10.1109/MMBIA.2001.991698.

[177] A. Yezzi, L. Zöllei, and T. Kapur. A variational framework for integrating segmentation and registration through active contours. *Med. Img. Anal.*, 7(2):171–185, 2003. doi:10.1016/S1361-8415(03)00004-5.

[178] S. Yoon, A. R. Pineda, and R. Fahrig. Level set reconstruction for sparse angularly sampled data. In *Proc. Nucl. Science Symp. and Med. Imag. Conf.*, vol. 6, pp. 3420–3423. IEEE, Oct. 2006. doi:10.1109/NSSMIC.2006.353736.

[179] D. F. Yu and J. A. Fessler. Edge-preserving tomographic reconstruction with nonlocal regularization. *IEEE Trans. Med. Imag.*, 21(2):159–173, 2002. doi:10.1109/42.993134.

[180] H.-K. Zhao, T. Chan, B. Merriman, and S. Osher. A variational level set approach to multiphase motion. *J. Computational Physics*, 127(1):179–195, 1996. doi:10.1006/jcph.1996.0167.