

OBJECT-BASED 3-D RECONSTRUCTION
OF ARTERIAL TREES
FROM A FEW PROJECTIONS

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Jeffrey Allen Fessler
August 1990

© Copyright by Jeffrey Allen Fessler 1999
All Rights Reserved

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Albert Macovski
(Principal Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Dwight Nishimura

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Thomas Binford

Approved for the University Committee on Graduate Studies:

Dean of Graduate Studies

Abstract

This dissertation describes an object-based approach to the problem of reconstructing three-dimensional descriptions of arterial trees from a few angiographic projections. The method incorporates *a priori* knowledge of the structure of branching arteries into a natural optimality criterion that encompasses the entire arterial tree. This global approach enables reconstruction from a few noisy projection images. We present an efficient optimization algorithm for object estimation, and demonstrate its performance on simulated, phantom, and *in vivo* magnetic resonance angiograms, as well as an X-ray phantom.

The 3-D reconstruction method is based on new nonparametric smoothing algorithms that we present for both linear and nonlinear measurements. These algorithms provide nonparametric alternatives to the Kalman filter and the extended Kalman filter. In particular, we describe automatic procedures based on cross-validation for determining how much to smooth; this adaptation allows the data to “speak for itself” without imposing a parametric model.

Acknowledgements

It has been a rewarding experience to have been able to address an applied problem of practical importance, while frequently digressing into the more theoretical world of splines. For this I am deeply grateful to my advisor, Al Macovski, for his support, patience, and kindness, and for the latitude that he granted me in my doctoral pursuits. I thank Dwight Nishimura for his thoughtful questions, and for his invaluable assistance with the magnetic resonance aspects of my work. I would also like to thank Tom Binford for assenting to read this dissertation, John Gill for his helpful ideas on computational issues, and Iain Johnstone for introducing me to splines and to Bernard Silverman, whose course on nonparametric regression successfully converted me to spline methods.

I am particularly indebted to Yoram Bresler, who patiently answered my incessant questions during my first two years at Stanford, and has continued to serve as my sounding board for crazy new ideas.

The MR results in this dissertation would have been impossible without the assistance of several people (to whom I collectively owe dozens of favors), including Dwight Nishimura, Doug Noll, Craig Meyer, John Pauly, Peter Webb, and especially Sam Wang. It was nice to have Steve Conolly available for bouncing ideas around; he also kindly read and edited several chapters of this thesis, thereby sparing the reader from a plethora of obfuscations. I also owe thanks to Craig Barrett for his clever graphics software.

A great many other friends have enriched my time at Stanford, but there is no need to list them here since they know better than to read this, especially my jazz musician colleagues.

Finally, I would like to express my thanks to my families; to my parents for their love and encouragement, and to Sue Cutler for her patience during my hectic race to finish this dissertation less than a week before our wedding. It is to her that I dedicate this work.

July 1990

Jeff Fessler

This work was supported by a National Science Foundation Graduate Fellowship, National Institute of Health contract NO1-HV-38045 and grant R01-HL-39045, National Science Foundation contracts ECS-8213959 and ECS88-01708, General Electric Medical Systems Group contract 22-84, and by National Institute of Health grants HL-39297, HL-34962, and HL-39478.

Contents

Abstract	iv
Acknowledgements	v
List of Tables	x
List of Figures	xi
1 Introduction	1
1.1 Why Quantitative Angiography?	1
1.2 Background	2
1.2.1 Single-View Methods	4
1.2.2 Two-View Methods	6
1.2.3 Multiple-View Methods	7
1.3 The New Method	9
1.4 Outline	11
I Object-Based Reconstruction from Projections	12
2 Measurement Model	13
2.1 Parallel X-ray Geometry	14
2.2 Parallel MR Geometry	17
2.3 Point-source X-ray Geometry	19
2.4 Blurring, Sampling, and Noise	20
2.5 Summary	22
2.6 Appendix: Discrete Blur Approximation	23

3	Object Model	24
3.1	SGC object model	24
3.2	Branching object model	26
3.3	Summary	29
3.4	Appendix: Ellipse Projection	29
3.5	Appendix: Projection of Intersecting Ellipses	31
3.6	Appendix: SGC to GC conversion	33
4	Single Object Estimation	34
4.1	Problem Setup	34
4.1.1	Projected Parameters	35
4.1.2	Projection Measurements	37
4.2	Maximum Likelihood Criterion	37
4.3	Gauss-Markov Criterion	38
4.4	Nonparametric Optimality Criterion	39
4.5	Smooth Background	40
4.6	Summary	42
5	Single Slice Estimation	43
5.1	Superimposed Signals	44
5.2	Estimate-Maximize Algorithm	44
5.3	Alternate-Maximize Algorithm	47
5.4	Initialization	49
5.5	Number of Views	49
5.6	Optimal Projection Angles	50
5.7	Conclusion	53
5.8	Appendix: Slow EM Convergence	54
6	3-D Object Estimation	56
6.1	Maximum Likelihood Criterion	56
6.2	Nonparametric Optimality Criterion	56
6.3	Parallel-Projection Estimation Algorithm	59
6.4	Cone-Beam Reconstruction Algorithm	62
6.4.1	Cone-Beam Projections of a SGC	63
6.4.2	Parameter Estimate Updates	65
6.5	Summary	66

7	Experimental Results	67
7.1	Simulation	67
7.2	Crescent Cross-Section Simulation	70
7.3	MR Phantom Data	71
7.4	MR Carotid Arteries	73
7.5	X-ray Phantom Data	75
II	Nonparametric Smoothing Algorithms	77
8	Linear Nonparametric Smoothing	78
8.1	Linear Smoothing	78
8.2	Spline Smoothing of Scalar Measurements	80
8.3	Spline Smoothing of Vector Measurements	83
8.4	Choosing the Smoothing Parameters	84
8.4.1	Unbiased Risk	85
8.4.2	Cross Validation	85
8.4.3	Generalized Cross-Validation	86
8.5	Error Covariance Estimation	86
8.6	Algorithm	87
8.7	Simulation Results	87
8.8	Summary	90
8.9	Appendix A: Spline Smoothing Derivation	90
8.10	Appendix B: CV Score Derivation	92
9	Nonlinear Nonparametric Smoothing	93
9.1	Introduction	93
9.2	Linearization Approach	96
9.3	Hessian Approach	97
9.4	Choosing the Smoothing Parameters	99
9.5	Algorithm	99
9.6	Edge Estimation Application	103
9.7	Summary	106
10	Discussion	107
10.1	Summary of Contributions	107
10.2	Open Problems	108

List of Tables

6.1	Iterative object estimation algorithm.	61
6.2	Starting rows ($n_{b,k}$) of objects in CGR phantom (see Figure 7.20).	63
6.3	Ending rows ($n_{e,k}$) of objects in CGR phantom (see Figure 7.20).	63
7.1	Imaging parameters for the data sets used to evaluate the reconstruction method.	67
7.2	RMS parameter estimation errors for eight iterations.	68
8.1	Boundary conditions for spline smoothing.	82
8.2	Computational requirements of linear smoothing algorithm.	88
8.3	Computational requirements for band matrix operations.	88
8.4	Relative efficiencies of the different smoothing approaches.	90
9.1	Iterative nonlinear estimation algorithm and computational requirements.	101

List of Figures

1.1	Two ellipses with different eccentricities and densities that have identical biplane projections.	7
2.1	Projection geometry, showing projections of a bifurcation.	14
2.2	MRSIR k -space trajectory and timing diagram.	18
3.1	A GC (left) and an approximating SGC (right).	25
3.2	Convention defining content attribute of a cross-section formed by two intersecting ellipses with different densities.	27
3.3	Two equivalent object sets. Note that although $\mathcal{Z}_2 \neq \mathcal{Z}_3$, $\mu_{\{\mathcal{O}_1, \mathcal{O}_2\}} = \mu_{\{\mathcal{O}_1, \mathcal{O}_3\}}$.	28
3.4	Example demonstrating the four ways arteries can terminate.	28
3.5	Projections of an elliptical cross-section. l_θ : ideal, $s_{\theta,i}$: blurred and sampled.	29
3.6	Line integral through overlapping ellipses.	31
4.1	Single circular artery and single view geometry.	34
4.2	Simulated projection of a synthetic artery, SNR=1.1 to 2.0.	37
4.3	Four example ML fits, measurements: (\cdot) , true: (solid), estimated: (dashed).	38
4.4	Projections of ML estimates of cross-sections from Figure 4.2.	38
4.5	Nonparametric radius estimates from Figure 4.2, true: (solid), estimated: (dashed).	40
4.6	Nonparametric position estimates from Figure 4.2, true: (solid), estimated: (dashed).	40
4.7	Nonparametric radius estimates from Figure 4.2, true: (solid), estimated: (dashed), with $\alpha_2 = 0$.	40
4.8	ML radius estimates from Figure 4.2, true: (solid), estimated: (dashed).	41
4.9	Simulated projection of a synthetic artery with unknown background added.	41
4.10	Nonparametric radius estimates from Figure 4.9, true: (solid), estimated: (dashed).	41
4.11	Nonparametric position estimates from Figure 4.9, true: (solid), estimated: (dashed).	41
5.1	Overlapping projections of multiple ellipses.	44
7.1	Simulated angiograms; $\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$.	68

7.2	True (above) versus estimated (below) stenosis cross-sections; $n = 160, \dots, 180$.	69
7.3	True (above) versus estimated (below) bifurcation cross-sections; $n = 130, \dots, 141$.	69
7.4	True (above) versus estimated (below) bifurcation cross-sections; $n = 80, \dots, 94$.	69
7.5	True (solid) versus estimated (dashed) object radii.	69
7.6	Radius estimation errors for the simulated objects.	69
7.7	3-D surface display of reconstructed simulated arteries; $\theta = 0^\circ$.	69
7.8	3-D surface display of reconstructed simulated arteries; $\theta = 45^\circ$.	69
7.9	Simulated angiograms measurements, $\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$.	70
7.10	True cross-sections (top) versus one-ellipse estimates (bottom).	70
7.11	True cross-sections (top) versus two-ellipse estimates (bottom).	70
7.12	MR tube-phantom projections; $\theta = 22^\circ, 67^\circ, 112^\circ, 157^\circ$.	71
7.13	Histogram of radius estimation error.	71
7.14	Reprojection of estimates (solid line) on an unused view; $\theta = 0^\circ$.	71
7.15	3-D surface display of reconstructed tube phantom; $\theta = 45^\circ$.	72
7.16	3-D surface display of reconstructed tube phantom; $\theta = -45^\circ$.	72
7.17	MR <i>in vivo</i> carotid angiograms; $\theta = 0^\circ, 45^\circ, 108^\circ, 143^\circ$.	73
7.18	3-D surface display of reconstructed carotid arteries; $\theta = 143^\circ$.	74
7.19	3-D surface display of reconstructed carotid arteries; $\theta = -50^\circ$.	74
7.20	GE-CGR X-ray phantom projections; $\theta = 18^\circ, 72^\circ, 108^\circ, 138^\circ$.	75
7.21	Radius estimation errors for the GE-CGR X-ray phantom.	75
7.22	Reprojection of estimates (solid line) on an unused view; $\theta = 0^\circ$.	75
7.23	Reprojection of estimates (solid line) on an unused view; $\theta = 48^\circ$.	76
7.24	Shaded surface of estimate; $\theta = 72^\circ$.	76
7.25	Shaded surface of estimate; $\theta = 238^\circ$.	76
8.1	True curves (-) and noisy measurements ($\circ, *$).	87
8.2	Comparison of MSE, UR, CV, and GCV for scalar measurements.	87
8.3	True curves (solid) and estimated curves (dashed) using CV score.	89
9.1	Noisy image data for curved edge estimation example.	102
9.2	Noisy image data for straight edge estimation example.	102
9.3	Comparison of MSE and CV_0 for curved edge example.	104
9.4	Comparison of MSE and CV_0 for straight edge example.	104
9.5	True (solid) and estimated (dashed) edge position from Figure 9.1.	105
9.6	True (solid) and estimated (dashed) edge position from Figure 9.2.	105
9.7	Estimation errors for edge estimation examples.	105

Chapter 1

Introduction

1.1 Why Quantitative Angiography?

A tremendous amount of research on imaging methods for the circulatory system¹ has been motivated by this astounding statistic [1]: “coronary heart disease causes one third to one half of all deaths of people between the ages of 35 and 64 years in the United States.” The objectives of this research were summarized by Skorton [2]:

the ultimate goals of clinical cardiac imaging include the complete structural, functional, and metabolic characterization of the heart, great vessels, and the pulmonary vasculature in a noninvasive manner. To achieve these goals will require the depiction of cardiac morphology ... and the anatomy of the coronary arteries.

Accurate descriptions of arterial trees would be useful for quantitative diagnosis of atherosclerosis, for planning treatment of stenotic² arteries, for monitoring disease progress or remission, and for evaluating efficacy of different treatments [3]. This dissertation describes a new approach to the problem of reconstructing three-dimensional descriptions of arterial trees from a few angiographic projection images.

The conventional method for evaluating angiographic images is human interpretation. The inaccuracy of human interpretation of angiograms has been well documented, in terms of both intra-observer and inter-observer variability [4]. Not only is the variability large, but also “the standard approach to assessing the severity of coronary stenoses—estimation of percent diameter narrowing of the stenotic segment compared to a presumably normal segment—may not correlate with the physiological significance of the obstruction [2].” This is due in part to the fact that “the hemodynamic effect³ of coronary artery narrowing is determined by

¹Broadly interpreted, the term *angiography* includes any such methods.

²A stenosis is a narrowing of a vessel cross-section.

³The decrease in fluid pressure across a stenosis.

the absolute (not relative) diameter and also by the length of the stenotic segment [4].” Furthermore, “since hemodynamic effects are proportional to the diameter of the stenosis raised to the fourth power, a small uncertainty in border definition may introduce a large uncertainty into the hemodynamic effect calculated from X-ray measured dimensions [5].” These issues are compounded by the fact that “many [radiologists] use [percent stenosis] without specifying whether they are referring to a percentage reduction of the diameter or of the area of the lumen⁴ [6].” These problems have motivated the development of automated methods for obtaining quantitative measurements of arterial morphology, because, as concluded by Paulin, “measurements from angiograms can be performed more objectively with advanced imaging technology assisted by computers [6].”

Current X-ray angiography procedures are applied only to patients with a high risk of having atherosclerosis. This is insufficient, as a summary of recent studies [7] discussed:

a fascinating but disturbing fact: ... asymptomatic subjects with normal results on stress tests had a much lower mortality rate than asymptomatic subjects with abnormal results. However, the very large low-risk group with normal responses to exercise testing ... contained, in absolute terms, the large majority of subjects who subsequently died suddenly.

This observation highlights the importance of non-invasive quantitative angiography, since asymptomatic patients do not qualify for invasive examinations such as intra-arterial angiography. To identify individuals at possible risk due to mild plaque build-up, a non-invasive procedure is needed that can quantify lumen size accurately. Magnetic resonance angiography (MRA), in conjunction with computed reconstructions of arterial trees, has potential to serve this need⁵. The 3-D arterial reconstruction algorithm presented in this dissertation is particularly suited to MR data, though the approach is also applicable to digitally acquired X-ray angiograms.

1.2 Background

Quantitative angiography has been studied for over two decades, and improvements in imaging technology and computer speed continue to spur the development of various reconstruction methods for the many different applications. To the best of our knowledge, all previous methods have been applied only to X-ray angiograms, and have not addressed the unique aspects of MRA. In this section, we review the quantitative angiography literature. For additional reviews and bibliographies, see [9, 10, 11, 12, 13, 14, 15].

For completeness, we first mention some alternate methods for improving the utility of angiography. One approach is to combine multiple projection images to form a focussed image of a plane or surface containing

⁴The lumen is the inner portion of an artery through which blood flows.

⁵The detectability of mild atherosclerosis remains to be seen. In a recent study [8], Glagov *et al.* “conclude that human coronary arteries enlarge in relation to plaque area and that functionally significant stenosis may be delayed until the lesion occupies 40 percent of the internal elastic lamina area.”

the arteries of interest, while blurring objects in other planes. This procedure is known as tomosynthesis [16, 17]. Another approach is to acquire a 3-D data set “directly.” Flow-sensitive MR techniques can acquire a complete 3-D data set [18, 19], but they require a large number of excitations. This requirement makes cardiac gating impractical, and ungated sequences often suffer from sensitivity to non-uniform flow and vessel motion. Specialized X-ray equipment [20, 21] with multiple source-detector pairs can acquire enough projection data to reconstruct a 3-D data set with conventional algorithms, but such equipment is not widely available. For quantitative diagnosis, one must still apply an algorithm to extract the pertinent information from such data sets [22, 23, 24, 21].

A so-called “reconstructed arterial tree” is not an actual arterial tree, but rather some symbolic description of the tree. The nature of the description has profound implications for the applicability and stability of a reconstruction method. The more rigorous of the methods described below are explicitly based on either a parametric or a nonparametric description of the arterial tree. Unfortunately, many papers invoke only an implicit model, such as those in which “diameter” is computed from a single view without mentioning the assumption of a circular lumen cross-section.

The most general representation for an arterial tree is to consider it an unknown three-dimensional density function $\mu(x, y, z; t)$. In general, μ is also a function of time due to the motion of blood and of the arteries. In X-ray imaging, μ represents the linear attenuation coefficient of iodinated blood, while in subtraction MRA, μ represents the density of inverted spins that have flowed into the readout region [25, 26]. We use “contrast” or “density” throughout to refer to μ , although no contrast agent is administered in MRA. Our goal is to estimate μ from as few views as possible.

Since the time-variation of arteries makes acquisition of only a few (almost) simultaneous views possible with practical equipment, the number of views required by a reconstruction method affects its clinical feasibility significantly. Thus, our review is categorized into single-view methods, stereo and bi-plane methods, and multiple-view methods.

1.2.1 Single-View Methods

Before the proliferation of digital angiography units, methods for quantitative angiography were based primarily on digitized film [27, 28, 29, 30]. Most film-based methods considered only a single view, perhaps because multiple-view methods require calibration of a 3-D reference coordinate system—a cumbersome procedure with film.

The earliest methods for processing angiograms were based on applying gradient operators, such as the Sobel and Hueckel operators [31], followed by thresholding to identify edges. Wiener-like filters for edge estimation were also applied [32]. To avoid missing any edges, the thresholds were set fairly low, leading to many false-alarm edges. The detected edges were then linked to form objects and to prune spurious edges [33, 34]. The difficulty with global image operators is that the intensity of an arterial projection can vary with the artery's diameter, so a single threshold is insufficient. Furthermore, gradient operators are sensitive to noise, and fail to exploit any properties of the arterial images of interest.

The inadequacy of edge detection methods led to the development of methods designed specifically for estimating the projected arterial diameter. A very popular approach has been to smooth the arterial profile in each scan line, and then use the peak of the first derivative or the zero of the second derivative as an estimate of the arterial edge [35, 36, 37]. However, these slope-based methods are biased, since an arterial profile is a semi-ellipse function rather than a rectangular function. (The first-derivative peak underestimates [29] and the second-derivative zero overestimates [35] arterial diameter.) Kooijman *et al.* attempted to combat this problem by using a weighted average of the two derivatives [28], but the weighting must be determined empirically. A line-by-line Wiener filter was also proposed for vessel edge detection [38].

The above methods are termed “geometric,” while methods that are based on the absolute measured pixel values are termed “densitometric.” For example, Weber [39] showed that the first zero-crossing of the Fourier transform of a vessel profile is proportional to arterial diameter and is somewhat insensitive to image blurring. Shmueli [40, 10, 41] first proposed fitting the projection of a circle to the measurements as a parametric method for estimating arterial diameter. Kruger [42] showed that vessel diameter is proportional to the ratio of the sum of the profile pixel values to the center pixel value. Such methods rely on a linear relationship between the X-ray path length through an iodinated vessel and the measured pixel value. The nonlinearity of film-based systems perhaps explains why the geometric methods remain popular, despite the fact that Pappas has shown that the densitometric approaches are more accurate [30, 35, 12]. This accuracy is due in part to the ability of densitometric methods to estimate diameter to sub-pixel resolution, while slope-based methods are limited by the detector resolution.

The simplest densitometric operation is accumulating the pixel values across an arterial profile. The resulting sum is proportional to the cross-sectional area. Unfortunately, the constant of proportionality is unknown and is difficult to determine since it depends on the concentration of iodine in X-ray angiography and on the density of excited spins in MRA. Furthermore, this density varies along the arterial length in MRA.

The unknown constant limits the utility of densitometric area to consideration of relative values. However, computations of stenosis pressure drop depend on stenosis geometry, including entrance and exit angle [43], for which relative area alone is insufficient.

Most of the above methods for estimating arterial diameter originally required manual tracing of the projected artery's centerline. Many papers have proposed methods for eliminating this interaction. Shmueli [41] developed an optimal estimation algorithm for computing the position and diameter of a single artery. Nishimura [44] generalized this to include overlapping arteries. Barth [27] described a single-vessel tracking algorithm based on a circular search window, while Hoffmann [45] used a pair of rectangular search windows to track an entire arterial tree recursively. Kitamura [14] described another tracking algorithm that accounts for bifurcations, but not overlapping arterial projections. Sun [46] exploited position, curvature, diameter, and density continuity to track an artery, but without accommodation of branching or overlapping arteries.

Sun [46] observed that there are two classes of methods for arterial segmentation: "tracking" methods that attempt to follow the local ridge of an arterial projection, and "scanning" methods that are applied more globally. Nguyen [47, 48] proposed one scanning method that requires several steps for ridge detection and merging. Stansfield [49] reported a edge-linking region-growing method controlled by a rule-based system, but concluded that the rules were inadequate. An expert-system based multi-resolution algorithm was also presented by Ergener [50]. Eichel [51] described a completely automated edge-tracking algorithm based on a Markovian edge model. After enhancing the edges with a Gaussian-weighted gradient operator, the edges are linked with a tree search based on communication theory. Morphological filters have recently shown promise for segmentation problems, and Besson [52] combined these filters with region growing and a "snake transform" to produce a segmentation algorithm that is well suited to angiograms. Another matched-filter/thresholding scheme was reported by Chaudhuri [53] for retinal reflectance images. Most scanning segmentation methods simply produce binary-valued images that distinguish arterial pixels from background pixels. To be useful, these methods must be followed by a method to compute the parameters of interest, such as arterial diameter.

All single-view methods are limited by the fact that the projected diameter can be misleading for eccentric cross-sections. This is a significant problem, since, as Brown reported [4], "the diseased lumen is commonly eccentric in cross-section." Another limitation is that distinguishing between overlapping, crossing, and bifurcating arteries is very difficult from only one view. These problems are important motivations for multiple-view methods.

1.2.2 Two-View Methods

There have been two motivations for considering two-view methods over one-view methods: 1) a better estimate of the lumen dimensions can be computed from two views, and 2) if the corresponding projections of an arterial segment can be identified in the two views, then the 3-D location of the artery can be computed. These two motivations have had a one-to-one correspondence with the following two paradigms for quantification: 1) process each pair of image rows to estimate arterial lumen, possibly accounting for some correlation between adjacent lumens, and 2) first process the two views individually to extract arterial segments, and then match corresponding segments between views.

In an early paper, Chang [54] discussed reconstruction of a binary matrix (representing lumen cross-section) from two orthogonal projections, and showed that the reconstruction was unique only for convex objects symmetric about both projection directions. Gerbrands [55] and Reiber [56] also used binary matrices to represent a single artery's cross-sections, but they also included a cost function to quantify the similarity of adjacent cross-sections, in hopes of overcoming the non-uniqueness for asymmetric lumens. Any method based on binary discretizations of cross-sections requires knowing or calibrating the constant of proportionality between the X-ray path length and the measured pixel intensity. Reiber [56] used a "normal" segment with an elliptical cross-section for this calibration. Fenster [57] used a binary matrix to approximate a crescent-shaped lumen within a circle whose radius is assumed known, and demonstrates reconstructions from two orthogonal simulated projections. These binary methods have had mixed reviews; Suetens [58] found that it was "next to impossible to reconstruct any part of a blood vessel."

Brown *et al.* [4] used the second view to obtain a better estimate of the cross-sectional area by using the expression: $\text{area} = \frac{\pi}{4}d_1d_2$, where d_1 and d_2 were hand-traced projected diameters in the two views. This formula is correct for an elliptical cross-section with axes of length d_1 and d_2 , but is incorrect if the ellipse axes are not aligned with the projection directions. Spears [59] calculated the error due to assuming aligned ellipse axes. Despite the potential error, several investigators have made this assumption [4, 60, 43, 36]. Kitamura [61] claimed that although the densitometric information in two views does not determine an ellipse uniquely, there are only two possible ellipses, and proposed arbitrarily choosing the more circular one. Furthermore, if the two projections are orthogonal, then Kitamura claimed that only the orientation of the two possible ellipses differs, so the area is uniquely determined. These statements are true only if the contrast density is known, as Figure 1.1 demonstrates.

For some neurosurgical applications, estimates of arterial centerlines and relative areas may suffice, and algorithms have been published for obtaining those features from just two views [62, 14, 63]. These algorithms usually require two steps: tracking arteries in each view, and then identifying corresponding arterial segments between the two views. Gerbrands [64] used a region growing algorithm to generate an arterial skeleton in each view, converts the skeleton to a graph with nodes at the bifurcations, and applies a minimum-cost tree-matching procedure to identify corresponding bifurcations in the two views. Kim [65]

Figure 1.1: Two ellipses with different eccentricities and densities that have identical biplane projections.

used the iodine bolus edge as the feature for matching segments. Suetens [58] exploited a consistency property motivated by consideration of the human stereopsis by matching arterial segments with similar orientation, width, and intensity. The correspondence problem is considerably easier with stereo angiograms since the difference in projection angle is small. Hoffmann [66] correlated image rows to compute the shift that determines arterial depth in stereo projections. Parker [62] entered landmarks in each view by hand, and matched segments between views using a dynamic programming algorithm. Kitamura [14] used manually identified segment correspondences. Venaille [67] proposed acquiring a third projection to reduce the number of consistent matches between arterial segments. Each of the detectors in most bi-plane systems can acquire a time sequence of N arterial projections within the duration of the iodine bolus. Wu [63] identified corresponding segments in all $2N$ ungated projections, and then used a weighted average of the $2N$ densitometric areas. Nevertheless, Wu [63] concluded that obtaining more reliable measurements will require “developing algorithms which include information from more than two projection views.”

1.2.3 Multiple-View Methods

The arterial descriptions that have been computed from two views have usually consisted of only arterial position and cross-sectional area. Though adequate for some applications, this hardly meets Skorton’s goal of a complete characterization of cardiac morphology. For applications where very accurate quantification of stenosis geometry is critical, one can justify acquiring a few views. The additional views should allow one to overcome the limitations of two-view methods by accommodating overlapping arterial projections and bifurcations.

To circumvent the ill-conditioned nature of reconstruction from just a few views, virtually all methods attempt to reduce the dimensionality by considering some special class of densities. The least restrictive class of descriptions considers μ to be a discretized 3-D voxel set. Jiya [68] described an algebraic reconstruction technique (ART) requiring three orthogonal views for reconstructing a 3-D voxel representation of symmetric objects, but with no consideration of noise. Spears *et al.* [69] used a maximum entropy algorithm for reconstructing a single asymmetric cross-section from three to nine projections. Consideration of a single object is unrealistic for multiple-view methods, since it is very unlikely that one can acquire more than one or two overlap-free views of an artery.

Recently, investigators have recognized that ill-conditioned inverse problems such as limited-view reconstruction need to be *regularized* by incorporating *a priori* knowledge about the structures of interest. Haneishi *et al.* [70] used two properties for regularization: 1) the background pixels and the arterial pixels (at least ideally) have different values, and 2) arterial pixels are connected. They incorporated these properties

into a penalized-likelihood optimality criterion, and used a simulated-annealing optimization algorithm. They demonstrated reconstructions from five to twenty simulated projections. Rougee [71] combined the sparseness, non-negativity, and connectedness properties of arterial trees into a detection-estimation scheme based on detecting a region of support, and compared it with ART and with the extreme-value technique (EVT). She reconstructed an X-ray phantom from fifteen views. Trouset [72] reduced the computational cost of this method by introducing a multiscale implementation, and demonstrated reconstruction of a phantom from thirty-two views.

Though the voxel-based methods have the ability to represent any arterial shape, accurate reconstruction seems to require several views. To reduce the number of views required while maintaining robustness to low SNR, stronger assumptions are necessary. As observed by Rossi [73], “the ultimate goal of processing the projection measurements is typically far more modest than obtaining high resolution cross-sectional imagery.” In fact, the goal is typically to obtain quantitative descriptions of arterial shape (perhaps as an intermediate step towards the goal of evaluating hemodynamic properties). Therefore, rather than making a futile attempt to reconstruct an arbitrary cross-section, Rossi [74, 75, 73] proposed using a parametric model for objects, and analyzed reconstruction of a circular disk from a few projections. A parametric object model reduces the number of unknowns, by exploiting our *a priori* knowledge of the structure of arteries, and translates the reconstruction problem into an estimation problem. Pappas [12] later demonstrated the accuracy of using elliptical cross-sections to represent arteries, since ellipses can better approximate a wider class of cross-sections than can circles. Rossi [76] further evaluated ellipse reconstruction accuracy. Bresler [77, 78] and Kitamura [14] both used generalized cylinders to represent a set of ellipses describing an artery.

The elliptical model captures *a priori* knowledge about the shape of arterial cross-sections in a simple parametric form. The equally important *a priori* knowledge that arteries are smooth, i.e., that ellipse parameters vary slowly along an arterial segment, is more difficult to quantify. Shmueli [41] and Bresler [78] used explicit stochastic Gauss-Markov models to parametrically quantify this smoothness. Using this model, Bresler [77, 79, 80, 78] presented an optimal minimum mean-squared error (MMSE) algorithm for reconstructing a single object with elliptical cross-sections from a few projections. Realizing that a single-object algorithm is impractical, Bresler also presented a suboptimal algorithm for reconstructing several disjoint objects from a few views, based on a hierarchical divide-and-conquer approach [81, 15]. In a joint effort, Bresler and this author demonstrated the hierarchical algorithm [82] on a set of four simulated projections of a few synthetic objects. Despite its suboptimality, the hierarchical algorithm performed nearly as well as the MMSE algorithm in reconstructing a single synthetic object from four simulated projections [83].

1.3 The New Method

This dissertation presents a new method for object-based reconstruction from a few views that overcomes several limitations of the previous work. In particular, the novel aspects of the method we present are:

- it is based on a global criterion—to maintain accuracy at the low SNR typical of non-invasive methods,
- bifurcating arteries are explicitly modeled, there are no empirically determined thresholds,
- overlapping vessel projections are accommodated,
- and the time-dependence of contrast density is modeled.

The new method is a mixture of parametric and nonparametric methods. We use a parametric model (ellipses) for arterial cross-sections, but a nonparametric method (splines) for arterial smoothness. Although the previously used parametric smoothness models allowed the derivation of theoretically satisfying MMSE estimators, there are two problems with that approach: 1) the parameters of a Gauss-Markov model are unknown and not easily determined, and 2) in general, these models imply that the *a priori* covariances of the cross-sections vary along the length of an artery. This implied variation is inconsistent with our intuition: prior to examining an angiogram, our uncertainty about cross-sectional shape is uniform along the arteries. We instead propose a nonparametric smoothing approach, described in Chapter 6, that captures our *a priori* knowledge of arterial smoothness with minimal assumptions. Bresler eventually came to agree with this approach [84]:

Parametric methods are useful when parsimony is important, as is the case when the measurements are few (e.g. limited projection angles), since the number of estimated parameters must be smaller than the number of measurements. However, we typically acquire many measurements along the length of an artery, so a nonparametric method for smoothness gives the greatest flexibility. Hence, mixed parametric/nonparametric methods make perfect sense when the degrees of freedom in the problem can be split into two sets: one which is well observed, and one which is poorly observed.

A significant limitation of the hierarchical algorithm [81, 15, 82] is that it was based on disjoint objects. Since a single ellipse is inadequate for representing bifurcations, where stenoses frequently occur, the hierarchical algorithm was applied only to simulated data. This dissertation describes a new extension of the generalized-cylinder object model that overcomes this limitation, allowing the first application of a parametric reconstruction method to *in vivo* projections of branching arteries.

Several algorithm developers have termed their methods “optimal,” and in some cases this may be inadvertently misleading. For example, Fleagel *et al.* [13] report a method for vessel border estimation based on

a multi-step process: an artery's projection is resampled perpendicular to its medial axis, a derivative operator is applied to each row of the resampled data, and a graph-searching technique is used to estimate the left and right vessel edges from the derivative values. Although the graph searching is performed optimally, this method is suboptimal overall because the derivative operation is an irreversible process. Thus, by the data-processing inequality [85], the estimated vessel boundaries are less accurate than a method that uses the original pixel values. The approach presented in this dissertation is based on a global optimality criterion that encompasses both the entire arterial tree and all of the measured data.

1.4 Outline

This thesis is organized in two parts. In Part I (Chapters 2 to 7), we describe the models and methods for object-based 3-D reconstruction. The methods are based on new nonparametric estimation algorithms that are detailed in Part II. These algorithms are of independent interest and have other applications, so Chapters 8 and 9 are self contained.

A reconstruction method is optimal only if the underlying models hold; in Chapter 2, we present a statistical model relating the projection images to the unknown arterial tree, and in Chapter 3, we describe a new object model, tailored for representing arterial trees. In Chapter 4, we apply the object and measurement models to the single-artery/single-view estimation problem. In Chapter 5, we consider algorithms for estimating a set of ellipses given their projections, and discuss selection of the projection angles. In Chapter 6, we propose a new optimality criterion for the problem of estimating an arterial tree from noisy projections, and describe the estimation algorithm. In Chapter 7, we show the results of applying this algorithm to simulated angiograms, to MR phantom data, to MRA projections of carotid arteries, and to X-ray phantom projections. Chapters 8 and 9 describe the nonparametric smoothing algorithms that are the theoretical core of this thesis. (The reader may find it useful to scan these chapters before reading Part I.) We discuss the results and the future directions of this research in Chapter 10.

Part I

Object-Based Reconstruction from Projections

Chapter 2

Measurement Model

Reconstruction from projections belongs to the mathematical class called *inverse* problems. Solutions to such problems depend heavily on the assumptions one makes about the corresponding *forward* processes. In this chapter, we present a statistical model for projection angiography that relates an unknown arterial tree to its acquired measurements. This model accounts for overlapping arterial projections, space-invariant blurring, and additive noise. In the next chapter we will restrict our attention to a specific class of arterial tree descriptions, but for this chapter we represent an arterial tree by an arbitrary three-dimensional density function.

A projection is a mapping of some physical property of a three-dimensional object into a 2-D image, and that mapping is determined by the acquisition geometry. This dissertation considers the cylindrical geometry shown in Figure 2.1, where the projection planes are rotated about a common axis, defined to be the z axis. The symbols u and v denote the local coordinates in the projection planes (so the z and v axes are parallel). MR systems can generate projections in arbitrary directions, and a more general model could exploit this versatility. However, X-ray systems are usually restricted to a single rotation axis, which motivates the cylindrical geometry used here.

Figure 2.1: Projection geometry, showing projections of a bifurcation.

In X-ray imaging the property of interest is the linear X-ray attenuation coefficient. The attenuation coefficient of blood does not differ significantly enough from that of soft tissue for blood to be visualized, so a bolus of iodine (contrast agent) is injected into the arteries immediately prior to imaging. In MR imaging, the pertinent property is the density of excited hydrogen spins. The contribution of flowing blood is differentiated from static tissue by exciting a region containing blood that is about to flow into the volume of interest. Note that in both modalities it is only the flowing blood that is imaged, so strictly speaking we will be reconstructing descriptions of the lumens of arterial trees.

Our goal is to develop a measurement model that is useful for both X-ray and MR angiograms. Although it may occasionally seem pedantic, we carefully state all the assumptions and approximations used while developing the model; these approximations are rather different for X-ray and MR imaging, so they are derived separately. Virtually all of the assumptions made in this chapter will have consequences for the results shown in Chapter 7.

The density function for an arterial tree varies with time, and a reconstruction paradigm that requires multiple views must account for this variation. Although one can minimize the variation of arterial tree position by using cardiac gating, the contrast will still vary from view to view as iodine (or excited spins) flow through the arteries. The object-based approach can easily accommodate this variation, as we discuss in the next chapter. We pay particular attention to the time sequencing of MR and X-ray angiography methods, since these differ considerably. These details are necessary since multiple-view methods inherently require more data acquisition time than single-view methods.

2.1 Parallel X-ray Geometry

For simplicity, we begin by considering a fictitious X-ray system with a plane wave of X-ray energy photons impinging on the object of interest¹. X-ray photons are absorbed and scattered as they pass through tissue, though for this section we assume the scattered photons are eliminated by collimation. The energy from unabsorbed X-ray photons is converted into a signal by an imaging chain. In a fluoroscopic system, this chain typically includes a scintillating phosphor screen, an image intensifier, and a television camera. The curved surface of an image intensifier introduces an image warping known as pincushion distortion [4]. Our first assumption is that this distortion has been corrected.

Assumption 2.1 *Pincushion distortion eliminated.*

¹Though fictitious, this parallel geometry becomes increasingly accurate and useful as the ratio of the object size to the source-to-object distance decreases.

Correcting this distortion usually requires acquiring images of a calibration grid, since pincushion distortion “cannot be described by a simple analytic function.” [28] Once this distortion is corrected, the projection mapping is expressed as parallel line-integrals through the density, according to the following form for the mean output of an imaging chain on the opposite side of the object from the source [86]:

$$I(u, v; t; \theta) = I_0 \int \eta_s(E) \eta_d(E) \exp \left[- \int \mu_l(s \cos \theta + u \sin \theta, s \sin \theta - u \cos \theta, v; t, E) ds \right] dE,$$

where I_0 is the incident source intensity. The functions $\eta_s(E)$ and $\eta_d(E)$ are the normalized source spectrum and the detector efficiency at photon energy E respectively, i.e., $\int \eta_s(E) dE = 1$. The term $\mu_l(x, y, z; t, E)$ represents the “total” linear X-ray attenuation coefficient corresponding to the sum of the contribution of the iodinated arteries of interest (μ) and the background tissue (μ_b), i.e.:

$$\mu_l(x, y, z; t, E) = \mu(x, y, z; t, E) + \mu_b(x, y, z; t, E).$$

This acquisition is known as the “live” image.

Assumption 2.2 *Snapshot acquisition.*

We assume each projection (or possibly pair of projections in bi-plane systems) is acquired very quickly, thereby “freezing” the density momentarily. Assume we acquire P projections, at time instants t_1, \dots, t_P and at projection angles $\theta_1, \dots, \theta_P$. For example, with a dual bi-plane DSA system [87] we could have: $P = 4, t_1 = t_2, t_3 = t_4, \theta_1 = 0^\circ, \theta_2 = 90^\circ, \theta_3 = \theta_r,$ and $\theta_4 = \theta_r + 90^\circ$. The snapshot assumption allows us to drop the explicit dependence of I on t , yielding:

$$I(u, v; \theta_p) = I_0 \int \eta_s(E) \eta_d(E) \exp \left[- \int \mu_l(s \cos \theta + u \sin \theta, s \sin \theta - u \cos \theta, v; t_p, E) ds \right] dE.$$

Assumption 2.3 *Monoenergetic source:*

$$\eta_s(E) = \delta(E - E_0).$$

The consequence of this assumption has been carefully evaluated by Simons [88, 89], who observed: “if one assumed a linear relationship between logarithmically subtracted video density and iodine density, only a small inaccuracy resulted.” Under this assumption, we can drop the dependence on photon energy:

$$I(u, v; \theta_p) = I_0 \eta_d(E_0) \exp \left[- \int \mu_l(s \cos \theta + u \sin \theta, s \sin \theta - u \cos \theta, v; t_p) ds \right].$$

The contribution of background tissues is significant in general, and cannot be ignored. Two methods have been used to circumvent this problem. At least conceptually, the simplest method is to acquire additional projection images of the background only (before the iodine injection or after the bolus has diluted), and

to perform logarithmic subtractions as discussed below. One potential problem with this approach is that any patient motion between the acquisitions will produce artifacts due to incomplete subtraction. This problem is germane to both single and multiple-view procedures, and several “rubber-sheet” algorithms for warping the background image to match the iodinated image have been proposed [90, 91]. A multiple-view approach may require moving the X-ray gantry to acquire the additional views, and then repositioning the X-ray gantry to acquire the background images. Since many angiographic systems are not precisely repositionable, the background image might not quite align with the corresponding iodinated image, which introduces another possible error source. The other compensation method is to just use the unsubtracted images and to approximate the background by global [12] or local [92, 14, 87], low-order polynomials, whose coefficients are estimated. This approach is also not without its problems: the background signal may be poorly approximated by low-order polynomials near bone edges.

Assumption 2.4 *Repositional system.*

If the X-ray system can be accurately repositioned, say, after the iodine bolus has diluted, then we can acquire another set of projections at angles $\theta_1, \dots, \theta_P$ at times $t_{1,b}, \dots, t_{P,b}$ of the background tissues only:

$$I_b(u, v; \theta_p) = I_0 \eta_d(E_0) \exp \left[- \int \mu_b(s \cos \theta + u \sin \theta, s \sin \theta - u \cos \theta, v; t_{p,b}) ds \right].$$

(Note that this implicitly assumes the source intensity I_0 is repeatable as well.) This acquisition is known as the “mask” image.

One could relax this assumption by using a separate smaller iodine injection for each view, and acquiring both the live and the mask images before rotating the X-ray gantry. However, this may lengthen the imaging procedure, thereby increasing the likelihood of patient motion.

Assumption 2.5 *Background synchronization:*

$$\mu_b(x, y, z; t_{p,b}) = \mu_b(x, y, z; t_p).$$

If the acquisitions are cardiac gated, and possibly respiratory gated, then the only differences between the background or “mask” image $I_b(u, v; \theta_p)$ and the iodinated image $I(u, v; \theta_p)$ are due to the contribution of the iodinated blood. Thus a logarithmic subtraction:

$$l_{\theta_p}(u, v; \mu) \triangleq \log I_b(u, v; \theta_p) - \log I(u, v; \theta_p),$$

yields the structure of interest:

$$l_{\theta_p}(u, v; \mu) = \int \mu(s \cos \theta_p + u \sin \theta_p, s \sin \theta_p - u \cos \theta_p, v; t_p) ds. \quad (2.1)$$

In this final form, $l_{\theta_p}(u, v; \mu)$ is the ideal projection of μ at angle θ_p and at time t_p . We postpone discussion of blurring and sampling until Section 2.4.

2.2 Parallel MR Geometry

The MR images discussed in Chapter 7 are based on the selective inversion-recovery (SIR) method [25]. “SIR bears some similarities to X-ray digital subtraction angiography (DSA). While DSA involves the subtraction of two images, one with and the other without contrast agent, SIR involves the subtraction of two images, one with and the other without inversion excitation of blood prior to its entering the region of interest.” [93] By using a multiple-readout selective inversion-recovery (MRSIR) sequence [26], we can acquire arterial views at a few projection angles without using invasive contrast agents or ionizing radiation. For a four minute MRSIR sequence, the signal-to-noise ratio is inversely related to the number of projections acquired, which necessitates optimal use of the projection data.

Most MR imaging takes advantage of the fact that the resonant frequency of a hydrogen proton in a magnetic field is proportional to the strength of the field. Spatially varying magnetic fields (gradients) are used to introduce an encoding of position into temporal frequency, so that at any given time during the acquisition interval, the received signal corresponds to a sample of the spatial-frequency content of the object being imaged. If one acquires enough samples of the spatial-frequency content, an image can be reconstructed.

Linear gradients are most commonly used to encode position into frequency, and any deviation from linearity produces a warping of the image that is somewhat analogous to pincushion distortion in an X-ray image intensifier. The analogy is functional in the sense that in both modalities one can image a rectangular-grid phantom to measure the geometric distortion and then “unwarp” other images. This was not necessary for the system (GE Signa) used to acquire the results of Chapter 7, but it may be necessary for other systems. Thus, the following assumption is inherent in our derivation:

Assumption 2.6 *No geometric distortion.*

The excited spins emit a radio-frequency (RF) signal that is received by an RF coil. An RF coil receives signals from the entire volume (within its sensitive region) with no directional selectivity, so there is a volume integral inherent in the system. Thus, if gradients are used to encode frequency only along x and z , then all the spins along y will contribute signal in unison, effecting a line-integral along y . Therefore, the line-integrals in an MR projection image are truly parallel. The basic imaging equation for a projection MR signal is:

$$s(t) = \iint \left[\int \mu(s \cos(\theta) + u \sin \theta, s \sin \theta - u \cos \theta, v; t) ds \right] e^{-jk_x(t)u} e^{-jk_y(t)v} du dv.$$

where $j = \sqrt{-1}$ and the spin density μ is now weighted by the spatial sensitivity of the receiver coil. We assume this sensitivity varies slowly over space. The k -space (spatial-frequency space) trajectories $k_x(t)$ and $k_y(t)$ are determined by the imaging sequence.

Assumption 2.7 *Negligible dephasing in a projection voxel.*

Figure 2.2: MRSIR k -space trajectory and timing diagram.

The spin density μ is in fact a complex quantity for MR angiography, due especially to dephasing induced by an inhomogeneous field. The effect can be minimized by using short echo-times [93], by selectively projecting only a slab containing the branches of interest [26] (rather than the entire volume), and by shimming the main field carefully [94]. Thus, we assume the dephasing is negligible, so that μ can be considered a real quantity. It may be necessary to generalize our results to a complex μ for systems with poor homogeneity. We also minimize velocity-dependent phase effects by using projection-reconstruction selective inversion-recovery (PRSIR) imaging [95, 96] and by using offset-echo acquisitions [93]. These considerations are part of the motivation for the PRSIR method, which may become the preferred sequence if the imaging time can be reduced by using multiple-readouts.

Assumption 2.8 *Effective extraction of real-valued images.*

The received signal is nevertheless complex, and we use a homodyning method [97, 98] to extract the in-phase component of interest. We assume that the necessary assumptions for homodyning [98] are met, including the assumption of slowly varying image phase. Though the phase in the 3-D volume may vary slowly, the projection of an arterial tree may superimpose two arteries of different phase, leading to signal loss. An interesting area for future work would be to use the reconstructed arterial tree to obtain a better phase reference under the assumption that the phase varies slowly along an artery’s length.

Since an MR readout measures only a sample of the spatial-frequency content of the selected region at a given time instant, the “snapshot” assumption used in the previous section needs careful examination. Full detail would be beyond the scope of this thesis, but the basic time sequence for MRSIR is displayed in Figure 2.2. At time A, an ECG R-wave triggers an inversion pulse that tags blood about to flow into the region of interest. To maximize wash-in for the carotid arteries, this tagging occurs just before the peak flow rate. At time B, about 400 ms later (during diastole), the first k -space line of the first projection is read. The duration of this readout is limited by the gradient strength and SNR considerations, and takes about 6 ms on our system, which brings us to time C. A 21 ms wait brings us to time D, when different gradients are used to readout the first k -space line of the second projection. This is repeated for all P projection images, so that after $400 + P * 27$ ms, we arrive at time E. At some later time F, another R-wave triggers an identical sequence of readouts—only this time no inversion pulse is used. There are typically 256 samples in each row of k -space, and the above procedure is repeated 128 times to acquire 128 rows in k -space.

To form projection images, the corresponding projection data (with and without inversion) are then subtracted and Fourier transformed. This subtraction should form images of the spins that flowed into the readout region. The following assumptions are implicit in this subtractive method:

Assumption 2.9 *No motion of “static” tissue.*

If there is no such motion, then the difference between the inverted and uninverted readouts will be due only to the flowing material.

Assumption 2.10 *Consistent wash-in and periodic gating.*

The readout for each row in k -space is triggered by a different heart beat. If the heart beats were very irregular and different amounts of spins were washed-in after each inversion pulse, then the acquired rows of k -space would correspond to the spatial-frequency content of *different* images, and there would be artifacts in the reconstructed images. Furthermore, the static tissues would have different T_1 relaxation periods, leading to imperfect subtraction.

Assumption 2.11 *Negligible wash-in during a readout.*

Similarly, if significant wash-in occurred during a readout, then that row of k -space would be inconsistent, also leading to artifacts. These artifacts are minimized by using the shortest readouts possible.

Of these assumptions, the last is perhaps the most questionable. We have often observed significant differences in contrast between the different projection images. There are two potential sources for these differences: more contrast in the later images due to additional wash-in, and errors in the tip-angle of the RF excitation, leading to different signal strengths. If the differences in contrast between images are due in part to wash-in, then the contrast almost certainly is changing during a readout, since the readout interval is a significant fraction of the time between readouts.

Provided all these assumptions hold, then the same basic projection relationship (2.1) holds for MR projections. We consider deviations from these assumptions in Section 2.4.

2.3 Point-source X-ray Geometry

In this section, we generalize the parallel-projection measurement model (2.1) by developing an approximate point-source projection model. Since the rays that pass from a point source to an area detector form a solid cone, this geometry is known as the cone-beam geometry. In addition to the assumptions of Section 2.1, we add the following:

Assumption 2.12 *Ideal point X-ray source.*

Real X-ray sources are of finite size and are tilted with respect to the detector plane [86]. This introduces additional blurring that we lump into the blur model of the next section. From [86], the ideal projection function due to a point source is (cf. (2.1)):

$$I_{\theta_p}^*(u, v; \mu) = \sqrt{1 + \frac{u^2 + v^2}{d^2}} \int \mu \left(s \cos \theta + \frac{u}{M(s)} \sin \theta, s \sin \theta - \frac{u}{M(s)} \cos \theta, \frac{v}{M(s)}; t_p \right) ds,$$

where

$$M(s) = \frac{d}{d - d_0 + s} = \frac{d}{d - d_0} \cdot \frac{1}{1 + \frac{s}{d - d_0}}$$

is the position-dependent magnification, d is the source-to-detector distance, and d_0 is the distance from the center of rotation to the detector. Note that as $d \rightarrow \infty$, $M(s) \rightarrow 1$.

Assumption 2.13 *Small centered object:* ($s \ll d - d_0$).

For objects that are close to the center of rotation, the magnification is approximately constant: $M(s) \approx M(0)$. Using this approximation, it is easy to see that

$$l_{\theta_p}^\bullet(u \cdot M(0), v \cdot M(0); \mu) \approx \int \mu(s \cos \theta + u \sin \theta, s \sin \theta - u \cos \theta, v; t_p) ds,$$

which is just a scaled version of the expression (2.1) for the parallel projection of μ . The accuracy of this approximation improves with increasing source-to-detector separation and with decreasing object breadth. Thus, by simply replacing Δ_h in (2.2) with $\Delta_h/M(0)$, we can apply our method to cone-beam projections, as we have done in Chapter 7. In Section 6.4 we discuss another method that may be useful for more accurate modeling and reconstruction from cone-beam projections.

2.4 Blurring, Sampling, and Noise

In the above sections, we derived expressions for ideal continuous projections. Actual imaging systems produce noisy, discrete samples of a blurred version of the ideal projections, and accurate estimation requires accounting for these distortions.

Angiographic images can contain significant high-frequency information due to the fine structure of small arteries. Since MR images are reconstructed from a finite number of spatial-frequency samples, they are virtually always undersampled. The resulting point-spread function (PSF) is determined by the frequency-space sampling and weighting, and is spatially invariant. In addition, any patient motion during the acquisition will introduce artifacts that may in part be modeled as blurring. In X-ray imaging, sources of blurring include the finite source and detector sizes and image-intensifier veiling glare. Although subtracting the mask image from the live image will eliminate X-ray scatter from the background tissues, any scatter from the iodinated arteries will remain. This object-dependent scatter also blurs the artery projection. Thus, in some situations it may be useful to estimate the PSF as well as the arterial tree [12, 14]. However, Pappas [12] has observed that even “if we assumed that there is no blurring, we would still get good estimates of the parameters.” Here, we assume the PSF is known,² spatially invariant, and denoted by $h(u, v)$.

Assumption 2.14 *Known, spatially invariant point-spread function.*

²In fact, for the X-ray phantom example of Chapter 7 that has an unknown PSF, we first used a local estimation algorithm similar to that of [12] to estimate the PSF, which was then held fixed for the global arterial tree reconstruction.

Digital imaging systems acquire samples of the blurred projections in the form of images. Let these images consist of N rows of W pixels each. If $s_{\theta_p, n, i}(\mu)$ denotes the ideal (mean) i^{th} pixel value in the n^{th} row of the projection image at angle θ_p , then for $i = 1, \dots, W$; $n = 1, \dots, N$; and $p = 1, \dots, P$:

$$s_{\theta_p, n, i}(\mu) \triangleq [h(u, v) ** l_{\theta_p}(u, v; \mu)] \Big|_{u=(i-i_h-\frac{1}{2})\Delta_h, v=z_n}, \quad (2.2)$$

where $z_n \triangleq (N - n)\Delta_v$, the horizontal and vertical pixel dimensions are Δ_h and Δ_v respectively, i_h is the distance (in pixels) from the projection of the rotation axis to the left edge of a projection image, and $**$ denotes the 2-D convolution operator. The offset i_h may differ from its usual value of $W/2$ since X-ray detectors are usually not perfectly centered on the rotation axis and since any extra linear phase in an MR system leads to a shift in the reconstructed images.

Blurring along the length of an artery's projection is less important for reconstruction than lateral blurring that smooths the artery's edges. In the next chapter we will restrict our attention to arteries whose medial axes are roughly parallel to the z axis, so it is reasonable to ignore blurring along this axis. This approximation simplifies the estimation algorithm discussed in Chapter 6.

Assumption 2.15 *Ignore vertical blurring: $h(u, v) = h(u)\delta(v)$.*

Assumption 2.16 *Smooth PSF.*

If the PSF is smooth, then we can approximate (2.2) by a discrete convolution that is more easily implemented. In the appendix we show that:

$$s_{\theta_p, n, i}(\mu) \approx h_i * l_{\theta_p, i}(z_n; \mu), \quad (2.3)$$

where $*$ now indicates discrete convolution with respect to i , and

$$h_i \triangleq h(i\Delta_h), \quad l_{\theta_p, i}(v; \mu) \triangleq \int_{(i-1)\Delta_h}^{i\Delta_h} l_{\theta_p}(u - i_h\Delta_h, v; \mu) du.$$

Assumption 2.17 *Independent, Gaussian distributed measurement errors.*

Finally, the actual measured pixel values are corrupted by noise. If $y_{p, n, i}$ denotes the measurement corresponding to $s_{\theta_p, n, i}$, then we assume

$$y_{p, n, i} = s_{\theta_p, n, i}(\mu) + \epsilon_{p, n, i},$$

where the additive measurement error $\epsilon_{p, n, i}$ has a Gaussian distribution with (possibly unknown) variance σ^2 . In MR imaging, the “source of this noise is thermally generated, randomly fluctuating noise currents in the body which are picked up by the receiving antenna,” [99] so it is correct to assume that the measurement errors are Gaussian and independent. With X-ray imaging, the Gaussian model is an approximation to the

Poisson noise distribution. The approximation improves with increasing photon count. One motivation for this assumption is that it leads to the computationally feasible algorithms of Chapters 8 and 9.

It is convenient to group the ideal projection and measurement samples in the different ways defined below. First group the samples by rows:

$$\mathbf{y}_{p,n} \triangleq [y_{p,n,1}, \dots, y_{p,n,W}]', \quad \mathbf{s}_{\theta_p,n}(\mu) \triangleq [s_{\theta_p,n,1}(\mu), \dots, s_{\theta_p,n,W}(\mu)]',$$

where “ $'$ ” denotes matrix transposition. Next group the corresponding rows of the P images:

$$\mathbf{y}_n \triangleq [\mathbf{y}'_{1,n}, \dots, \mathbf{y}'_{P,n}]', \quad \mathbf{s}_n(\mu) \triangleq [\mathbf{s}'_{\theta_1,n}(\mu), \dots, \mathbf{s}'_{\theta_P,n}(\mu)]'. \quad (2.4)$$

Finally, let

$$\mathbf{y} \triangleq [\mathbf{y}'_1, \dots, \mathbf{y}'_N]', \quad \mathbf{s}(\mu) \triangleq [\mathbf{s}'_1(\mu), \dots, \mathbf{s}'_N(\mu)]'$$

be the aggregates of all the samples. The vector \mathbf{y} has length NPW . We similarly define ε , yielding the final measurement model:

$$\mathbf{y} = \mathbf{s}(\mu) + \varepsilon, \quad (2.5)$$

which simply says that the (known) measurements are equal to the sampled and blurred projection of the (unknown) arterial tree density plus noise.

2.5 Summary

In summary, we have defined a statistical measurement model (2.1) (2.3) (2.5) that relates the unknown 3-D arterial tree density to the measured projections. The final form of our model is identical for both MR and X-ray angiography, but the assumptions involved are quite different.

It is worth noting that none of the assumptions used above are unique to our object-based approach. In fact, all paradigms must contend with inaccuracies that result from modeling error. Since our reconstruction approach uses global information, it clearly has the potential to be more robust to local deviations from the measurement models.

2.6 Appendix: Discrete Blur Approximation

In this appendix, we derive a simple approximation for discrete implementation of the blur convolution. Since we ignore the vertical blurring, we derive the approximation only in terms of the horizontal blurring. From (2.2), the expression for the ideal discrete sample is:

$$\begin{aligned}
 s_i &= [h(u) * l(u)]|_{u=(i-i_h-\frac{1}{2})\Delta_h} \\
 &= \int_{-\infty}^{\infty} l(\tau)h((i-i_h-\frac{1}{2})\Delta_h-\tau) d\tau \\
 &= \sum_j \int_{(j-1)\Delta_h}^{j\Delta_h} l(\tau)h((i-i_h-\frac{1}{2})\Delta_h-\tau) d\tau.
 \end{aligned}$$

If the PSF h is nearly constant over the interval $[(j-1)\Delta_h, j\Delta_h]$, then:

$$\begin{aligned}
 s_i &\approx \sum_j \int_{(j-1)\Delta_h}^{j\Delta_h} l(\tau)h(((i-i_h-\frac{1}{2})-(j-\frac{1}{2}))\Delta_h) d\tau \\
 &= \sum_j l_j h_{i-j},
 \end{aligned}$$

where

$$\begin{aligned}
 l_i &\triangleq \int_{(i-1)\Delta_h}^{i\Delta_h} l(\tau-i_h\Delta_h) d\tau, \\
 h_i &\triangleq h(i\Delta_h).
 \end{aligned}$$

Note that if the blur function is the discrete impulse response, i.e. $h_i = \delta_i$, then the ideal sample is the average of the signal intensity over the pixel size. This is a sensible upper bound on the performance.

Chapter 3

Object Model

As discussed in the introduction, the information provided by only a few projections is insufficient for reconstructing a general three-dimensional function. Any solution therefore hinges on restricting consideration to a smaller class of descriptions. In this chapter, we introduce an extension of the generalized cylinder (GC) object model, tailored to the task of representing arterial trees (the terms ‘object’ and ‘arterial segment’ are interchangeable). In particular, our model is designed to represent bifurcations accurately. This accuracy is essential since atherosclerotic lesions are prevalent near arterial branches [100, 101]. Binford [102, 103] introduced the GC model for computer vision applications. According to Agin [103], “A generalized cylinder consists of a space curve, or axis, and a cross-section function defined on that axis.” For this chapter, we take that cross-section function to be an ellipse.

3.1 SGC object model

A true GC would have arbitrarily oriented ellipses, as in Figure 3.1. Reconstruction of such general objects from projections is still an open problem; to our knowledge, the only work on this problem is a hierarchical algorithm proposed, but never implemented, by Bresler [15]. The comments of Marr [104, pp. 223–224] suggest that GC reconstruction may be challenging to implement from projections acquired around a single rotation axis. We instead adopt the popular approach [14, 78, e.g.] of parameterizing objects by their cross-sections parallel to the xy plane, an approach ideally suited to the cylindrical geometry. As argued in [14], a GC with slowly varying elliptical cross-sections can be approximated by a set of parallel ellipses as shown in Figure 3.1. Such a set of ellipses can be parameterized by z , hence we call the collection a single-valued generalized cylinder (SGC). Objects that wind back upon themselves (e.g. U shaped) must be represented by more than one SGC, and are called ‘multi-valued.’ The further an object is tilted away from the rotational axis, the poorer its SGC representation. Thus, as in most imaging procedures, proper patient positioning

Figure 3.1: A GC (left) and an approximating SGC (right).

is essential, and the arteries of interest should be aligned as close as possible to the rotational axis. The examples of Chapter 7 demonstrate successful reconstruction of objects with tilts exceeding 45° . Though we parameterize objects by parallel cross-sections, the cross-sections perpendicular to an artery’s medial axis are more important for quantitative diagnosis. Hence, we present formulae for converting between GC and SGC parameters in Section 3.6.

Each SGC cross-section has three attributes: position, shape, and content. An ellipse’s position attribute is parameterized by the xy coordinates of its center, denoted by (c_x, c_y) . We parameterize an ellipse’s shape attribute by its radius (geometric mean of long and short axes), eccentricity¹ (ratio of long to short axis²), and orientation, denoted by r , λ , and ϕ respectively (see Figure 3.5). The content attribute parameterizes the density within an artery, and can have a profound, though often underestimated, impact on a reconstruction algorithm’s accuracy and applicability. Some methods assume the content is known or is computable from “normal” sections of an artery, usually assumed to have a circular cross-section. Having more than two views allows us to relax these assumptions. We do not assume the contrast density is known, and we allow it to vary along the length of an artery, from artery to artery, and possibly even between projection images, as discussed next.

Though the measurement model (2.5) allows a general time-varying density, we now consider only SGCs whose position and shape attributes are invariant at times t_1, \dots, t_P , but we account for the time variation of the content attribute³. This approach is necessary since the contrast often varies between MRA projection images, and is approximately sufficient since cardiac gating can synchronize position and shape. We assume each particular arterial cross-section is uniformly filled with contrast agent, so we parameterize an ellipse’s content by a vector denoted by $\boldsymbol{\rho} = [\rho_1, \dots, \rho_P]'$, where P is the number of projections. For some imaging techniques, we may be able to equate some of the ρ_p ’s. For example, with bi-plane DSA, the contrast is identical for each pair of projection images, so we would assume $\rho_1 = \rho_2$ and $\rho_3 = \rho_4$. Note that ideal data sets would have $\rho_1 = \dots = \rho_P$, since estimating additional parameters often decreases estimation accuracy, but we have found the full generality presented here necessary for some MRA data.

We now define notation for a hierarchy consisting of ellipses, objects, and trees. The symbol \mathbf{x} will denote

¹The radius/eccentricity parameterization is preferable to the long-axis/short-axis parameterization for our approach, since independence of radius and eccentricity is a more realistic assumption, particularly in stenotic arteries.

²Another definition of ellipse eccentricity is $\sqrt{1 - \lambda^{-2}}$.

³We could also account for lateral vessel motion between acquisitions [87].

a particular set of $5 + P$ ellipse parameters. In particular,

$$\mathbf{x}_k(z) = \begin{bmatrix} c_x(z) \\ c_y(z) \\ r(z) \\ \lambda(z) \\ \phi(z) \\ \rho(z) \end{bmatrix}$$

denotes the (unknown) ellipse parameters of the k^{th} object at vertical position z . An object is uniquely determined by specifying its starting plane z_b , its ending plane z_e , and the collection of parameters of its elliptical cross-sections between those planes. We use the symbol \mathcal{O}_k to denote the k^{th} object, i.e.,

$$\mathcal{O}_k \triangleq \{\mathcal{Z}_k, \mathcal{X}_k\},$$

where

$$\mathcal{Z}_k \triangleq [z_{b,k}, z_{e,k}]$$

denotes the vertical domain of the k^{th} object, and

$$\mathcal{X}_k \triangleq \{\mathbf{x}_k(z) : z \in \mathcal{Z}_k\}$$

denotes the collection of ellipse parameters. We will say more about \mathcal{Z}_k later in this section.

An arterial tree is a collection of K objects, or ‘object-set,’ denoted by the symbol Ψ :

$$\Psi \triangleq \{K, \mathcal{O}_1, \dots, \mathcal{O}_K\} = \{K, \mathcal{Z}_1, \mathcal{X}_1, \dots, \mathcal{Z}_K, \mathcal{X}_K\}. \quad (3.1)$$

The notational hierarchy for Ψ directly corresponds to a hierarchical data structure [62] that we used to implement the algorithm described in Chapter 6. Due to the simplicity of the ellipse parameterization, this representation for Ψ is considerably more compact than a discretized 3-D voxel set. More importantly, the significant factors for quantitative diagnosis, such as percent stenosis and stenosis eccentricity, are directly computable from Ψ .

With these definitions, we have translated the problem of reconstructing a density μ from projections into the problem of estimating an object-set Ψ from projections. That is, we must estimate the number of objects, the vertical extent of each object, and the parameters of the ellipse cross-sections for each object.

3.2 Branching object model

In earlier efforts [83, 82], we assumed that the K objects to be reconstructed were disjoint. Although convenient, this assumption precluded accurate modeling of branching vessels. Histological sections of bifurcations [100, 101] suggest that pairs of intersecting ellipses can approximate cross-sections of a bifurcation

Figure 3.2: Convention defining content attribute of a cross-section formed by two intersecting ellipses with different densities.

accurately, even a diseased one. Therefore, we discard the disjointedness assumption by allowing SGC objects to intersect and by accounting⁴ for their intersections [105]. The importance of this accounting is clear from early GC-based vision algorithms [103] that were prone to failure near the intersection of GCs. (For simplicity, we consider bifurcations only; the extension to trifurcations involves only additional notation.)

When two SGC objects intersect, their position and shape attributes are unaffected, but we must specify their content attributes more carefully. Although it may be reasonable to assume that two intersecting ellipses must share the same density, we can simplify our reconstruction algorithm by not enforcing this constraint. This also provides a self-test for the reconstruction: if the estimated densities differ significantly for two arterial branches near a bifurcation, then something is clearly amiss. For mathematical consistency, we must specify a symmetrical definition for the content of two intersecting ellipses with different densities. Our convention is described pictorially in Figure 3.2, where we define the density of the common area of two ellipses to be the average of the two ellipse densities. If the two densities are the same, then the area of overlap will have that same density.

With the above definitions, we can now specify the unique 3-D density function generated by a given object-set. Let $\mathcal{E}(\mathbf{x})$ be the set of points within the ellipse parameterized by \mathbf{x} , i.e., if $\mathbf{x} = [c_x \ c_y \ r \ \lambda \ \phi]'$, then

$$\mathcal{E}(\mathbf{x}) = \left\{ (x, y) : \left(\frac{(x - c_x) \cos \phi - (y - c_y) \sin \phi}{r \sqrt{\lambda}} \right)^2 + \left(\frac{(x - c_x) \sin \phi + (y - c_y) \cos \phi}{r / \sqrt{\lambda}} \right)^2 \leq 1 \right\}. \quad (3.2)$$

The relationship between an object-set Ψ and its density μ_Ψ is then:

$$\begin{aligned} \mu_\Psi(x, y, z; t_p) &= \sum_{k=1}^K \rho_{k,p}(z) \mathbf{1}_{\mathcal{E}(\mathbf{x}_k(z))}(x, y) \mathbf{1}_{\mathcal{Z}_k}(z) \\ &- \sum_{k_1=1}^K \sum_{k_2=1}^K \frac{\rho_{k_1,p}(z) + \rho_{k_2,p}(z)}{2} \mathbf{1}_{\mathcal{E}(\mathbf{x}_{k_1}(z))}(x, y) \mathbf{1}_{\mathcal{E}(\mathbf{x}_{k_2}(z))}(x, y) \mathbf{1}_{\mathcal{Z}_{k_1}}(z) \mathbf{1}_{\mathcal{Z}_{k_2}}(z), \end{aligned} \quad (3.3)$$

where

$$\mathbf{1}_{\mathcal{Z}}(z) = \begin{cases} 1, & z \in \mathcal{Z} \\ 0, & z \notin \mathcal{Z} \end{cases}.$$

The first summation would suffice for a set of disjoint objects, and the second summation accounts for the areas of overlap.

⁴More perspiration than inspiration: compare Sections 3.4 and 3.5.

Figure 3.3: Two equivalent object sets. Note that although $Z_2 \neq Z_3$, $\mu_{\{\mathcal{O}_1, \mathcal{O}_2\}} = \mu_{\{\mathcal{O}_1, \mathcal{O}_3\}}$.

Figure 3.4: Example demonstrating the four ways arteries can terminate.

The reverse relationship to (3.3) is not unique because many different object-sets generate the same density function. One trivial reason for this non-uniqueness is that an ellipse's shape is the same for orientations ϕ and $\phi + \pi$, but the parameter vectors differ. Also, if one changed Ψ by adding objects with zero density, there would be no change in μ_Ψ . More significantly, the union of two identical ellipses is indistinguishable from a single ellipse. We will say two object-sets are *equivalent* if the density functions they generate from (3.3) are equal. For example, if the three objects shown in Figure 3.3 all have the same density, then the object-set formed by combining \mathcal{O}_1 and \mathcal{O}_2 is equivalent to the object-set formed by combining \mathcal{O}_1 and \mathcal{O}_3 . Technically then, we will be reconstructing an *equivalence class* of object-sets from projections.

Since a given object-set Ψ generates a unique density function μ_Ψ by (3.3), we can speak of the projection of an object-set, denoted by $\mathbf{s}(\mu_\Psi)$. Although \mathbf{s} is linear in μ , it is not linear in Ψ since μ is a nonlinear function of Ψ . In fact, since we are not assuming disjoint objects, \mathbf{s} is not even additive in general, i.e., the projection of an object collection differs from the sum of each object's projection. Nevertheless, we can write expressions for $\mathbf{s}(\mu_\Psi)$ in closed form using the formulae derived in Sections 3.4 and 3.5.

The non-uniqueness of an object-set actually works to our advantage since we need not estimate z_b or z_e exactly for objects near bifurcations, as demonstrated by Figure 3.3. What defines the endpoints of an arterial segment? There are four possibilities for SGC objects:

- an artery may leave the region of interest (ROI),
- an artery may 'fade-out' due to incomplete filling by contrast agent,
- an artery may be occluded or taper down to a size below the effective resolution,
- or an artery may connect to another artery (branching).

These four possibilities are illustrated in Figure 3.4. Of these, only the first possibility has a unique value of z_b (or z_e), and this value is easily determined. For the others, we can always conservatively make the object extra long, and let the density or radius become vanishingly small. For a multi-valued GC object, the endpoints of its SGC approximation are poorly defined, and are determined in practice by the manual-entry described in Chapter 6.

Figure 3.5: Projections of an elliptical cross-section. l_θ : ideal, $s_{\theta,i}$: blurred and sampled.

3.3 Summary

We have described an extension of the generalized-cylinder object model based on elliptical cross-sections. By allowing objects to intersect, the model can represent arterial trees accurately. This model defines the class of 3-D density functions considered, namely those that can be expressed in the form (3.3).

The following three sections derive mathematical expressions that are only necessary for implementing the reconstruction method described in Chapter 6.

3.4 Appendix: Ellipse Projection

Consider a collection of ellipses in a given plane, some of which may intersect. Since we restrict our model to bifurcations, the collection's projections are the superposition of the projections of those ellipses that are isolated, plus the sum of the projections of the pairs of intersecting ellipses. Thus we need two types of formulae: projections of a single ellipse, and projections of intersecting ellipses (cf. (6.8)). In this section and the next, we present expressions for these projections and their partial derivatives. These are needed by the nonlinear vector-spline smoothing algorithm described in Chapter 9, since it is a gradient-descent optimization method.

Consider a single isolated ellipse with parameters $\mathbf{x} = [c_x \ c_y \ r \ \lambda \ \phi \ \rho]'$. Since we consider projections only in the plane of this ellipse, we drop the dependence on z (and hence v and n). From (2.1), (3.2), and (3.3), the ideal continuous line-integral of an ellipse is:

$$\begin{aligned}
l_\theta(u; \mathbf{x}) &= \int \rho \mathbf{1}_{\mathcal{E}(\mathbf{x})}(s \cos \theta + u \sin \theta, s \sin \theta - u \cos \theta) ds \\
&= 2\rho h_p \sqrt{r_p^2 - (u - u_p)^2} \mathbf{1}_{|u - u_p| \leq r_p},
\end{aligned}$$

where

$$\begin{aligned}
u_p &\triangleq c_x \sin \theta - c_y \cos \theta, \\
r_p &\triangleq r \cdot \sqrt{a_+}, \quad h_p \triangleq a_+^{-1},
\end{aligned} \tag{3.4}$$

and

$$a_\pm = \frac{1}{2} \left((\lambda \pm \lambda^{-1}) - (\lambda \mp \lambda^{-1}) \cos(2(\phi - \theta)) \right).$$

Then from (2.2) and (2.3),

$$s_{\theta,i}(\mathbf{x}) = h_i * l_{\theta,i}(\mathbf{x}),$$

where

$$\begin{aligned}
l_{\theta,i}(\mathbf{x}) &= \int_{(i-i_h-1)\Delta_h}^{(i-i_h)\Delta_h} l_\theta(u; \mathbf{x}) du \\
&= \rho h_p \left[u \sqrt{r_p^2 - (u - u_p)^2} + r_p^2 \arcsin \left(\frac{u}{r_p} \right) \right] \Big|_{u=u_L}^{u=u_R},
\end{aligned}$$

where

$$u_L \triangleq \max \left\{ \frac{(i - i_h - 1)\Delta_h}{r_p}, -1 \right\}, \quad u_R \triangleq \min \left\{ \frac{(i - i_h)\Delta_h}{r_p}, 1 \right\},$$

for i in the range

$$\{i : |(i - i_h - \frac{1}{2})\Delta_h - u_p| \leq r_p + \frac{1}{2}\Delta_h\},$$

otherwise the values are 0. By the linearity of convolution, the partial derivatives of $s_{\theta,i}(\mathbf{x})$ are the blurred partial derivatives of $l_{\theta,i}(\mathbf{x})$, which are given below for i in the same range:

$$\begin{aligned}
\frac{\partial l_{\theta,i}(\mathbf{x})}{\partial c_x} &= -(\sin \theta)(l_\theta(u_R; \mathbf{x}) - l_\theta(u_L; \mathbf{x})), \\
\frac{\partial l_{\theta,i}(\mathbf{x})}{\partial c_y} &= (\cos \theta)(l_\theta(u_R; \mathbf{x}) - l_\theta(u_L; \mathbf{x})), \\
\frac{\partial l_{\theta,i}(\mathbf{x})}{\partial r} &= 2\rho r \left(\arcsin \left(\frac{u_R}{r_p} \right) - \arcsin \left(\frac{u_L}{r_p} \right) \right), \\
\frac{\partial l_{\theta,i}(\mathbf{x})}{\partial \lambda} &= \rho \frac{-a_-}{\lambda a_+^2} \left(\sqrt{r_p^2 - (u_R - u_p)^2} - \sqrt{r_p^2 - (u_L - u_p)^2} \right), \\
\frac{\partial l_{\theta,i}(\mathbf{x})}{\partial \phi} &= \rho \frac{-(\lambda - \lambda^{-1}) \sin(2(\phi - \theta))}{a_+^2} \left(\sqrt{r_p^2 - (u_R - u_p)^2} - \sqrt{r_p^2 - (u_L - u_p)^2} \right), \\
\frac{\partial l_{\theta,i}(\mathbf{x})}{\partial \rho} &= l_{\theta,i}(\mathbf{x})/\rho.
\end{aligned}$$

$\mathbf{s}(\mu_\Psi)$ could be implemented as a subroutine that accepts an object-set as its argument and returns a vector of length NPW . However, we achieve substantial memory and computational savings by exploiting the fact

Figure 3.6: Line integral through overlapping ellipses.

that projections of ellipses are semi-ellipse ‘bump’ functions whose supports are only small fractions of the size of the projection vector.

3.5 Appendix: Projection of Intersecting Ellipses

The projection of two intersecting ellipses is more complicated due to the region of overlap. Consider two intersecting ellipses with parameters $\mathbf{x}_1 = [c_{x,1} \ c_{y,1} \ r_1 \ \lambda_1 \ \phi_1 \ \rho_1]$ and $\mathbf{x}_2 = [c_{x,2} \ c_{y,2} \ r_2 \ \lambda_2 \ \phi_2 \ \rho_2]$, and let \mathcal{E}_1 and \mathcal{E}_2 denote the corresponding ellipse sets (3.2). For reasons that will become clear in Chapter 6, we are more interested in the signal that results from the difference of the projection of the union of the two ellipses and the projection of \mathbf{x}_2 . This signal is denoted by $s_{\theta,i}^*(\mathbf{x}_1)$, where the dependence on \mathbf{x}_2 is suppressed since it will be held fixed. The definition of this signal (cf. (6.8)) is:

$$s_{\theta,i}^*(\mathbf{x}_1) = h_i * l_{\theta,i}^*(\mathbf{x}_1),$$

where

$$l_{\theta,i}^*(\mathbf{x}_1) \triangleq \int_{(i-i_h-1)\Delta_h}^{(i-i_h)\Delta_h} l_{\theta}^*(u; \mathbf{x}_1) \, du,$$

and

$$l_{\theta}^*(u; \mathbf{x}_1) \triangleq l_{\theta}(u; \mathbf{x}_1, \mathbf{x}_2) - l_{\theta}(u; \mathbf{x}_2).$$

Applying (2.1) and (3.3) and using an overline to represent set complements:

$$\begin{aligned}
l_\theta^*(u; \mathbf{x}_1) &= \int_{\{(x,y):x \sin \theta - y \cos \theta = u\}} \left[\left(\rho_1 \mathbf{1}_{\varepsilon_1} + \rho_2 \mathbf{1}_{\varepsilon_2} - \frac{\rho_1 + \rho_2}{2} \mathbf{1}_{\varepsilon_1 \cap \varepsilon_2} \right) - \left(\rho_2 \mathbf{1}_{\varepsilon_2} \right) \right] dl \\
&= \int_{\{(x,y):x \sin \theta - y \cos \theta = u\}} \left[\rho_1 \mathbf{1}_{\varepsilon_1} \mathbf{1}_{\overline{\varepsilon_2}} + \frac{\rho_1 - \rho_2}{2} \mathbf{1}_{\varepsilon_1} \mathbf{1}_{\varepsilon_2} \right] dl \\
&= (\rho_1) f_1(d_\theta(u; \mathbf{x}_1), l_\theta(u; \mathbf{x}_1)/\rho_1, d_\theta(u; \mathbf{x}_2), l_\theta(u; \mathbf{x}_2)/\rho_2) \\
&\quad + \left(\frac{\rho_1 - \rho_2}{2} \right) f_2(d_\theta(u; \mathbf{x}_1), l_\theta(u; \mathbf{x}_1)/\rho_1, d_\theta(u; \mathbf{x}_2), l_\theta(u; \mathbf{x}_2)/\rho_2),
\end{aligned}$$

where for $j = 1, 2$:

$$d_\theta(u; \mathbf{x}_j) \triangleq c_{x,j} \cos \theta + c_{y,j} \sin \theta - (u - u_{p,j})(\lambda_j - \lambda_j^{-1}) \frac{1}{2a_{+,j}} \sin(2(\phi_j - \theta))$$

is the distance shown in Figure 3.5. The path length functions shown in Figure 3.6 are:

$$\begin{aligned}
f_1(d_1, l_1, d_2, l_2) &\triangleq \text{length}([d_1 - l_1/2, d_1 + l_1/2] \cap \overline{[d_2 - l_2/2, d_2 + l_2/2]}), \\
f_2(d_1, l_1, d_2, l_2) &\triangleq \text{length}([d_1 - l_1/2, d_1 + l_1/2] \cap [d_2 - l_2/2, d_2 + l_2/2]),
\end{aligned}$$

where

$$\text{length}([a, b] \cap \overline{[c, d]}) \triangleq \begin{cases} (b - a) - (d - c), & a \leq c, b \geq d \\ b - d, & a \geq c, b \geq d, a \leq d \\ c - a, & a \leq c, b \leq d, b \geq c \\ 0, & \text{otherwise} \end{cases},$$

and

$$\text{length}([a, b] \cap [c, d]) \triangleq \begin{cases} d - c, & a \leq c, b \geq d \\ d - a, & a \geq c, b \geq d, a \leq d \\ b - c, & a \leq c, b \leq d, b \geq c \\ b - a, & a \geq c, b \leq d \\ 0, & \text{otherwise} \end{cases}.$$

Our implementation uses the following approximation, obtained by interchanging the order of the convolution and the ‘length’ (min/max) operations:

$$\begin{aligned}
l_{\theta,i}^*(\mathbf{x}_1) &\approx (\rho_1) f_1(d_{\theta,i}(\mathbf{x}_1), l_{\theta,i}(\mathbf{x}_1)/\rho_1, d_{\theta,i}(\mathbf{x}_2), l_{\theta,i}(\mathbf{x}_2)/\rho_2) \\
&\quad + \left(\frac{\rho_1 - \rho_2}{2} \right) f_2(d_{\theta,i}(\mathbf{x}_1), l_{\theta,i}(\mathbf{x}_1)/\rho_1, d_{\theta,i}(\mathbf{x}_2), l_{\theta,i}(\mathbf{x}_2)/\rho_2),
\end{aligned}$$

where

$$d_{\theta,i}(\mathbf{x}) = d_\theta((i - i_h - \frac{1}{2})\Delta_h; \mathbf{x}).$$

From these formulae, we see that the partial derivatives of $\mathbf{s}^*(\mathbf{x})$ are combinations of those given for $l_{\theta,i}(\mathbf{x})$ in the previous section, and those of $d_{\theta,i}(\mathbf{x})$, given below:

$$\frac{\partial d_{\theta,i}}{\partial c_x} = \cos \theta + (\sin \theta)(\lambda - \lambda^{-1}) \frac{1}{2a_+} \sin(2(\phi - \theta)),$$

$$\begin{aligned}
\frac{\partial d_{\theta,i}}{\partial c_y} &= \sin \theta - (\cos \theta)(\lambda - \lambda^{-1}) \frac{1}{2a_+} \sin(2(\phi - \theta)), \\
\frac{\partial d_{\theta,i}}{\partial r} &= 0, \\
\frac{\partial d_{\theta,i}}{\partial \lambda} &= -((i - i_h - \frac{1}{2})\Delta_h - u_p) \sin(2(\phi - \theta)) / (\lambda a_+), \\
\frac{\partial d_{\theta,i}}{\partial \phi} &= ((i - i_h - \frac{1}{2})\Delta_h - u_p)(\lambda - \lambda^{-1}) \frac{a_-}{a_+^2}, \\
\frac{\partial d_{\theta,i}}{\partial \rho} &= 0.
\end{aligned}$$

3.6 Appendix: SGC to GC conversion

Our reconstruction algorithm provides estimates of the parameters of the elliptical cross-sections of a SGC, as shown in Figure 3.1. However, it is the variations in eccentricity, radius, and area perpendicular to an artery's axis that are important diagnostically. In this section, we present equations for converting between the parameters of a GC and a SGC. These relations are derived by approximating a SGC locally by an elliptical cone. The position attribute is identical for the GC and SGC models. To first order, the content attribute is also the same, since density usually varies quite slowly. Suppose the estimated shape parameters are $[r_S \lambda_S \phi_S]$, and that the local slope of the SGC is β_x and β_y . Let $\beta = \sqrt{1 + \beta_x^2 + \beta_y^2}$, then

$$r = \frac{r_S}{\sqrt{\beta}}$$

and

$$\lambda + \lambda^{-1} = \beta^{-1} \lambda_S + \beta \lambda_S^{-1} + (\beta - \beta^{-1})(\lambda_S - \lambda_S^{-1}) \sin^2(\phi_S - \arctan(\beta_y / \beta_x)).$$

Ellipse orientation is not well defined for an arbitrary GC axis [15] in general.

Chapter 4

Single Object Estimation

In this chapter, we describe several methods for quantifying a single artery given only a single projection image. (The problem of three-dimensional reconstruction of a single object from multiple views is just a special case of the algorithm to be presented in Chapter 6.) The single-view problem has been considered by numerous investigators, and is useful for comparative evaluation of new methods for quantitative angiography. Though a single-object algorithm has limited applications, the exposition gives us an opportunity to introduce the philosophy behind our 3-D reconstruction method in a much simpler setting.

The information provided by a single view is insufficient for determining the three shape parameters of an ellipse, so it is futile to consider an elliptical cross-section. Therefore, we restrict our attention in this chapter to arteries with circular cross-sections. It is also important to emphasize that the assumption of a parallel geometry is implicit in this chapter. The distance between an object and the detector is unavailable from a single view, so it is impossible to account for any depth-dependent magnification. One could use the image of the catheter as a scaling device, but this requires making the questionable assumption that the catheter and the artery lie in the same plane parallel to the detector.

4.1 Problem Setup

The geometry of this single-view problem is shown in Figure 4.1. This problem was first studied from an optimal estimation perspective by Shmueli [40, 41]. This perspective is the foundation for our work as well, but part of the purpose of this chapter is to relax four assumptions inherent in Shmueli's method: circular projected cross-sections, known fixed density, parametric Gauss-Markov model, and vessel discretization. Of

Figure 4.1: Single circular artery and single view geometry.

course, the most restrictive assumption is that of a single artery, which precludes overlapping projections and branches.

As shown in Figure 4.1, the measurements of an artery lie on a grid of N rows, each with W pixels. Each row of this image consists of samples of the projection of an arterial cross-section. Shmueli assumed that these *parallel* cross-sections of an artery were circles. However, it is clear from Figure 4.1 that if an artery with a circular cross-section is at all tilted, then the projected cross-section is actually an ellipse. Fortunately, the relationship between the true radius, denoted by $r(z)$ and the projected radius $r_p(z)$ depends only on the tilt parallel to the imaging plane, which is an observable.

Shmueli also assumed that the density of the artery was known and constant. This is unrealistic, particularly for MR images, so we include an additional parameter to represent the density, denoted by $\rho(z)$. The final parameter of interest is the x position of the artery, denoted by $\tau(z)$.

4.1.1 Projected Parameters

It is useful to note some relationships among the parameters, which follow from Sections 3.4 and 3.6. Let β_x and β_y denote the tilt of the artery with respect to the x and y axes respectively. The intersection of the artery with a plane that is perpendicular to the image and intersects the image at slice z is an ellipse with radius, eccentricity, and orientation denoted by r_e , λ_e , and ϕ_e , respectively, where

$$\begin{aligned} r_e(z) &= r(z) \sqrt[4]{1 + \beta_x^2(z) + \beta_y^2(z)}, \\ \lambda_e(z) &= \sqrt{1 + \beta_x^2(z) + \beta_y^2(z)}, \\ \phi_e(z) &= \tan^{-1}(\beta_y(z)/\beta_x(z)). \end{aligned}$$

These parameters are projected to form the semi-ellipse function shown in Figure 4.1, whose half-width and height are denoted by r_p and h_p , respectively, where

$$\begin{aligned} r_p(z) &= r_e(z) \sqrt{a_+(z)} = r(z) \sqrt{1 + \beta_x^2(z)} \\ h_p(z) &= \rho(z) 2r_e(z) / \sqrt{a_+(z)} = \rho(z) 2r(z) \sqrt{1 + \beta_x^2(z) + \beta_y^2(z)} / \sqrt{1 + \beta_x^2(z)}, \end{aligned} \tag{4.1}$$

where

$$a_+(z) = (1 + \beta_x^2(z)) / \sqrt{1 + \beta_x^2(z) + \beta_y^2(z)}.$$

Note that h_p depends on β_y , the tilt out of the plane, which is unobservable. However, β_x is observable, since

$$\beta_x(z) = \frac{\partial c_x(z)}{\partial z} = \dot{\tau}(z).$$

If we accumulated the pixel values in a row of a (noiseless) image, we would get the density-weighted area of the semi-ellipse:

$$A_p(z) = \rho(z) \pi r_e^2(z) = \rho(z) [\pi r^2(z)] \sqrt{1 + \beta_x^2(z) + \beta_y^2(z)}. \tag{4.2}$$

Note that this area consists of the true artery area πr^2 , scaled by the unknown density and by a factor that again depends on the unobservable β_y , which renders A_p essentially useless. This point has been overlooked repeatedly in the literature. For example, the percent area stenosis is often computed as follows:

$$\% \text{ stenosis} \stackrel{?}{=} \frac{A_p(z_{\text{stenosis}})}{A_p(z_{\text{“normal”}})},$$

i.e., by dividing the computed area in a stenotic segment by the area in a presumed normal segment. If $\beta_y(z_{\text{stenosis}}) \neq \beta_y(z_{\text{“normal”}})$, i.e. the y tilts are different at the two segments, then this percent-stenosis estimate will be incorrect.

Kruger [42] noticed that the true arterial radius is proportional to the semi-ellipse area divided by the peak projection value. The ellipse area can be estimated by summing a row of pixels, the peak can be estimated by choosing the maximum pixel value, and the slope can be estimated from the estimated arterial centerline, leading to the following simple estimate:

$$\hat{r}(z) = \frac{2}{\pi} \frac{\hat{A}_p(z)}{\hat{h}_p(z) \sqrt{1 + \hat{\beta}_x^2(z)}}.$$

The problem with this densitometric approach is that \hat{h}_p is very variable at low SNR. We prefer to estimate r_p and τ using the algorithms given below, and then to convert back to the object's radius using this expression (cf. (4.1)):

$$\hat{r}(z) = \frac{\hat{r}_p(z)}{\sqrt{1 + \hat{\beta}_x^2(z)}}.$$

Figure 4.2: Simulated projection of a synthetic artery, SNR=1.1 to 2.0.

4.1.2 Projection Measurements

The three parameters are of course continuous, but we observe only discrete, noisy samples of their projections (cf. (2.5)):

$$\mathbf{y}_n = \mathbf{s}(\mathbf{x}(z_n)) + \boldsymbol{\varepsilon}_n, \quad n = 1, \dots, N,$$

where

$$\mathbf{x}(z_n) \triangleq \mathbf{x}_n \triangleq \begin{bmatrix} \tau(z_n) \\ r_p(z_n) \\ \rho(z_n) \end{bmatrix}. \quad (4.3)$$

The projection function \mathbf{s} has components defined as follows:

$$\begin{aligned} s_i([\tau \ r \ \rho]') &\triangleq 2 \int_{i-1}^i \rho \sqrt{r^2 - (s - \tau)^2} \mathbf{1}_{\{|s - \tau| \leq r\}} ds \\ &= \left[r^2 \left(\kappa_+ \sqrt{1 - \kappa_+^2} + \arcsin \kappa_+ \right) - r^2 \left(\kappa_- \sqrt{1 - \kappa_-^2} + \arcsin \kappa_- \right) \right] \mathbf{1}_{\{|i - \tau - 1/2| \leq r + 1/2\}}, \end{aligned} \quad (4.4)$$

where

$$\kappa_- \triangleq \max \left\{ -1, \left| \frac{i - 1 - \tau}{r} \right| \right\}, \quad \kappa_+ \triangleq \min \left\{ 1, \left| \frac{i - \tau}{r} \right| \right\}.$$

Our objective is to estimate the parameters $\{\mathbf{x}_n\}_{n=1}^N$ from the measurements $\{\mathbf{y}_n\}_{n=1}^N$.

4.2 Maximum Likelihood Criterion

Now that we have defined an object model and a measurement model, the problem becomes one of estimation. The easiest approach would be simply to find the artery whose computed projections are the closest to the measurements in some sense. Under the AWGN assumption, the least-squares estimate is also the maximum-likelihood (ML) estimate, defined by:

$$\hat{\mathbf{x}}_{n,\text{ML}} \triangleq \arg \min_{\mathbf{x}} \|\mathbf{y}_n - \mathbf{s}(\mathbf{x})\|^2, \quad n = 1, \dots, N. \quad (4.5)$$

This criterion was studied extensively by Pappas [12], and was shown to be more accurate than methods based on finding zero-crossings of a profile's derivatives.

To evaluate the ML estimator, we created the synthetic projection image shown in Figure 4.2. These measurements were generated using (4.4) with added pseudo-random white Gaussian noise with variance $\sigma^2 = 16$. The resulting SNR ($\triangleq h_p/\sigma$) ranges from 1.1 to 2.0. For each row of this image, we computed the estimate $\hat{\mathbf{x}}_{n,\text{ML}}$ using criterion (4.5). To perform this nonlinear minimization, we used Powell's method [106], with the true parameters as the starting point. Despite this ideal initialization, the ML results are poor!

Figure 4.3: Four example ML fits, measurements: (\cdot), true: (solid), estimated: (dashed).

Figure 4.4: Projections of ML estimates of cross-sections from Figure 4.2.

Figure 4.3 compares four of the true profiles $\mathbf{s}(\mathbf{x}_n)$, the measured profiles \mathbf{y}_n , and the projected estimates $\mathbf{s}(\hat{\mathbf{x}}_{n,\text{ML}})$; the four were chosen at random. At least in this display, the results look reasonable. However, Figure 4.4 shows the projections of the estimates in image format, and the results look unacceptable. The difference in our interpretation of Figures 4.3 and 4.4 is due to the fact that the former shows only *local* properties, while the latter reveals *global* properties of the estimates, and the inconsistencies from line to line conflict with our concept of arterial smoothness. This variability is due to noise and the “threshold effect” of ML estimation [107] for low SNR.

Many researchers have attempted to introduce knowledge of smoothness “after the fact” by post-processing the ML estimates with various smoothing filters [27, 28, 11, 13, 55, 43, 66, 14]. The smoothing filter is usually chosen heuristically, with an empirically determined bandwidth. Such two-step approaches are suboptimal, and are not robust to the heavy tails of the ML estimate error at low SNR. Both the Gauss-Markov criterion and the nonparametric optimality criterion presented below attempt to incorporate directly our *a priori* knowledge of arterial smoothness into the estimation.

4.3 Gauss-Markov Criterion

Shmueli’s approach to accounting for arterial smoothness was to model the artery as a realization of a first-order discrete-time Gauss-Markov random process, i.e.:

$$\mathbf{x}_{n+1} = \mathbf{F}\mathbf{x}_n + \mathbf{G}\mathbf{u}_n,$$

where \mathbf{u}_n is normally distributed system noise with covariance \mathbf{Q} . Under this assumption, the maximum *a posteriori* (MAP) estimate of the states minimizes this criterion:

$$\sum_{n=1}^N \frac{\|\mathbf{y}_n - \mathbf{s}(\mathbf{x}_n)\|^2}{\sigma^2} + \sum_{n=1}^{N-1} (\mathbf{x}_{n+1} - \mathbf{F}\mathbf{x}_n)' (\mathbf{G}\mathbf{Q}\mathbf{G}')^{-1} (\mathbf{x}_{n+1} - \mathbf{F}\mathbf{x}_n).$$

The practical problem with this approach is that the parameters of this model (\mathbf{F} , \mathbf{G} , and \mathbf{Q}) are unknown and not easily determined. In addition, this Gauss-Markov model is an inherently discrete formulation, whereas an artery is continuous. The nonparametric criterion of the next section addresses both of these problems.

4.4 Nonparametric Optimality Criterion

Smoothing always involves a tradeoff between the conflicting goals of fit to the measurements and smoothness of the estimated functions. Nonparametric regression offers a solution to this tradeoff, and requires only an assumption of integrability of the square of the second derivative of the estimand. Most nonparametric regression literature has been concerned with estimating scalar functions from linear, scalar measurements. The natural generalization of these “penalized-likelihood” [108] or “regularized” [109] methods to our nonlinear, multi-dimensional, object estimation problem is the following estimator and optimality criterion¹:

$$\hat{\mathbf{x}}(\cdot) \triangleq \arg \min_{\mathbf{x}(\cdot)} \left[\sum_{n=1}^N \|\mathbf{y}_n - \mathbf{s}(\mathbf{x}(z_n))\|^2 + \boldsymbol{\alpha}' \int \ddot{\mathbf{x}}^2(z) dz \right], \quad (4.6)$$

where $\ddot{\mathbf{x}}(z)$ is the 3-dimensional vector containing the second partial derivatives of the components of $\mathbf{x}(z)$ with respect to z . The right-hand term in (4.6) quantifies arterial smoothness and prohibits excessive object wiggleness due to noise. The tradeoff is controlled by the smoothing factor $\boldsymbol{\alpha} = [\alpha_1 \ \alpha_2 \ \alpha_3]'$; for small $\boldsymbol{\alpha}$, the estimate fits the data closely, and for large $\boldsymbol{\alpha}$, the estimate becomes very smooth. Intuitively, we would use a fairly large value for α_1 since the arterial position varies relatively slowly, a very large value for α_3 since density typically varies very slowly, and a smaller value for α_2 to avoid oversmoothing the radius function. These qualitative statements are unsatisfying theoretically. Fortunately, nonparametric regression offers a solution: we can estimate $\boldsymbol{\alpha}$ from the data itself using cross-validation [110], as we discuss in detail in Chapters 8 and 9. In practice, essentially only α_2 changes with disease, and the other elements of $\boldsymbol{\alpha}$ could be fixed for a given imaging technique.

In Chapters 8 and 9 we present a computationally efficient algorithm for computing estimates of the form given by (4.6). Since it is a modified gradient-descent method, this algorithm requires that the user provide analytical derivatives of the projection function \mathbf{s} , which are given by:

$$\begin{aligned} \frac{\partial \mathbf{s}_i([\tau \ r \ \rho]')}{\partial \tau} &= -\rho 2r \left(\sqrt{1 - \kappa_+^2} - \sqrt{1 - \kappa_-^2} \right) \mathbf{1}_{\{|i-\tau-1/2| \leq r+1/2\}}, \\ \frac{\partial \mathbf{s}_i([\tau \ r \ \rho]')}{\partial r} &= 2(\arcsin(\kappa_+) - \arcsin(\kappa_-)) \mathbf{1}_{\{|i-\tau-1/2| \leq r+1/2\}}, \\ \frac{\partial \mathbf{s}_i([\tau \ r \ \rho]')}{\partial \rho} &= 2 \int_{i-1}^i \sqrt{r^2 - (s - \tau)^2} \mathbf{1}_{\{|i-\tau-1/2| \leq r+1/2\}}. \end{aligned}$$

Note that we do *not* apply any derivative operations to the noisy measurements.

To apply the nonparametric algorithm of Chapter 9 to the data shown in Figure 4.2, the only remaining requirement is to provide an initial estimate. For the case of a single artery this is simple to do automatically. We chose the following heuristic: a temporary copy of each row of the image was convolved with a “boxcar” kernel: [11111], (a crude matched filter), and the index of the pixel with maximum value in each row was

¹This criterion is nonparametric in the sense that we have avoided using a parametric (e.g. Gauss-Markov) model for the evolution of cross-section parameters along an artery. The cross-sections are still modeled parametrically.

Figure 4.5: Nonparametric radius estimates from Figure 4.2, true: (solid), estimated: (dashed).

Figure 4.6: Nonparametric position estimates from Figure 4.2, true: (solid), estimated: (dashed).

stored. This set of $N = 128$ numbers was then seven-point median filtered and the result was the initial position estimate. We arbitrarily initialized the radius to be 4 pixels based on visual image inspection; in a typical clinical setting the initial radius would be set to the normal size of the particular artery being studied. We initialized the density parameter to be 0.9, the sum of all of the image pixel values and divided by $N\pi 4^2$ (cf. (4.2)).

Figure 4.5 and Figure 4.6 are plots of the true and the estimated radius and position functions, where the smoothing parameters were chosen by using cross-validation². The RMS estimation errors for position and radius were 0.19 and 0.16 pixels, respectively. Such subpixel estimation accuracy justifies the computational effort of this global approach.

When using the cross-validation score to automatically select α , there is always some risk that the selected α_2 will be too large, thereby oversmoothing a stenosis. To prevent this occurrence, we can force α_2 to be zero, i.e., no smoothing of the radius function. Figure 4.7 shows the estimated radius function from Figure 4.2 with α_1 and α_3 set to the values chosen by cross-validation, but α_2 set to zero. For comparison, Figure 4.8 shows a plot of the radius estimated by the ML estimator. The nonparametric approach is clearly superior. Apparently, incorporating our *a priori* knowledge of the smoothness of the position and density functions is a significant improvement, even without making any assumptions regarding the smoothness of the radius function. In some respects, the estimated radius shown in Figure 4.7 may be visually preferable to the smooth radius shown in Figure 4.5 since the former gives one an impression of the variability of the estimates. If the variations were very large, then one might suspect that an apparent stenosis was just estimation error, but the decrease in radius shown in Figure 4.7 is clearly significant compared to the small variations, so we can be confident of the presence of a severe stenosis.

4.5 Smooth Background

Consider the simulated angiogram shown in Figure 4.9. This image is identical to Figure 4.2, except that an unknown background component has been added to the image. The change in intensity shown in Figure 4.9 is typical of image intensifier based systems.

²Specifically, we used the CV_0 score of Section 9.4.

Figure 4.7: Nonparametric radius estimates from Figure 4.2, true: (solid), estimated: (dashed), with $\alpha_2 = 0$.

Figure 4.8: ML radius estimates from Figure 4.2, true: (solid), estimated: (dashed).

Figure 4.9: Simulated projection of a synthetic artery with unknown background added.

As discussed in Section 2.1, one way to accommodate this background is to model it locally by a low-order polynomial [92, 12, 14, 87]. In this context, “locally” means across the artery’s profile, with perhaps a few pixels on either side. To estimate the coefficients of the polynomials, we need only augment the state vector:

$$\mathbf{x}(z_n) \triangleq \mathbf{x}_n \triangleq \begin{bmatrix} \tau(z_n) \\ r_p(z_n) \\ \rho(z_n) \\ \gamma_0(z) \\ \gamma_1(z) \end{bmatrix},$$

and redefine the projection function (cf. (4.4)):

$$s_i([\tau \ r \ \rho \ \gamma_0 \ \gamma_1]') \triangleq (\gamma_0 + i\gamma_1) \mathbf{1}_{\{|i-\tau| \leq r+2\}} + 2 \int_{i-1}^i \rho \sqrt{r^2 - (s-\tau)^2} \mathbf{1}_{\{|s-\tau| \leq r\}} ds.$$

The partial derivatives of this new projection function with respect to γ_0 and γ_1 are simply 1 and i respectively, so we can again apply the algorithm of Chapter 9. The smoothness of the background will be reflected in α_4 and α_5 , as determined by cross-validation.

Figures 4.10 and 4.11 compare the true and estimated position and radius parameters from Figure 4.9. Again the errors are all below a pixel. However, the RMS error for the position and radius were 0.29 and 0.27 respectively, which are slightly higher than the RMS errors found in the previous section (without the unknown background). Naturally, allowing the two extra degrees of freedom for the background has increased the estimation variance.

Figure 4.10: Nonparametric radius estimates from Figure 4.9, true: (solid), estimated: (dashed).

Figure 4.11: Nonparametric position estimates from Figure 4.9, true: (solid), estimated: (dashed).

4.6 Summary

We have presented three optimality criteria for the problem of quantifying a single artery given a single view: maximum likelihood, maximum *a posteriori* using a parametric Gauss-Markov model, and the new nonparametric approach (4.6).

Our formulation has been in terms of the state vector defined in (4.3). The second component of this vector is the *projected* radius. One could argue that smoothness should be expressed in terms of the *true* object radius, rather than the projected radius. Since the projected radius is related to the true radius and the derivative of the object's position, an alternative to (4.6) would be:

$$\hat{\mathbf{x}}(\cdot) = \arg \min_{\mathbf{x}(\cdot)} \left[\sum_{n=1}^N \|\mathbf{y}_n - \mathbf{s}_a(\mathbf{x}(z_n))\|^2 + [\alpha_1 \ 0 \ \alpha_2 \ \alpha_3]' \int \ddot{\mathbf{x}}^2(z) dz \right],$$

where

$$\mathbf{x}_n = \begin{bmatrix} \tau(z) \\ \dot{\tau}(z) \\ r(z) \\ \rho(z) \end{bmatrix},$$

and

$$\mathbf{s}_a([\tau \ \dot{\tau} \ r \ \rho]) \triangleq \mathbf{s}([\tau \ r / \sqrt{1 + \dot{\tau}^2} \ \rho]).$$

Since the projection function $\mathbf{s}_a(\mathbf{x})$ depends on $\dot{\tau}$, one would have generalize the algorithm of Chapter 9 to use this formulation. However, if the position function varies smoothly with z , then the transformation (4.1) that relates the true radius to the projected radius function will be smooth, so the projected radius function is approximately as smooth as the true radius function. Thus, the approach of Section 4.4 is adequate for SGC objects. An interesting extension of this work would be to consider the multi-valued case in a single view.

Chapter 5

Single Slice Estimation

Most automatic methods for 3-D reconstruction first process the individual 2-D views using algorithms similar to those of Chapter 4, and then attempt to identify corresponding structures from view to view. An interesting exception to that paradigm is the hierarchical algorithm proposed by Bresler [15]. As proposed, his method would have first estimated a collection of short 3-D object segments, and then combined them together using a Bayesian criterion. As we actually implemented it [83, 82], our algorithm first estimated a set of ellipses on a slice-by-slice basis and then attempted to combine ellipses from adjacent slices to form objects. Thus, as a complement to Chapter 4, in this chapter we compare two algorithms for estimating a collection of ellipses from a few noisy projections. The algorithms apply to parallel and cylindrical imaging geometries. These algorithms are presented only for completeness; we found the global approach described in Chapter 6 to be more effective, and the results in Chapter 7 used that method.

The two methods to be compared are the Estimate-Maximize (EM) algorithm and the Alternate-Maximize (AM) algorithm. These two algorithms originated in rather different places; the EM algorithm is rooted in statistics [111], while the AM algorithm is a simple method for nonlinear optimization [106, p. 310]. The ellipse estimation problem is a special case of the general problem of estimating the parameters of superimposed signals observed in additive white Gaussian noise, to which both the EM and the AM algorithms have been applied [112, 113]. Many other methods have also been considered [114, 115, 116]. The side by side comparison presented in this chapter provides insight into both algorithms.

In addition to the estimation algorithms, Section 5.5 discusses how many projection angles are needed for ellipse reconstruction, and Section 5.6 derives the optimal projection angles.

Figure 5.1: Overlapping projections of multiple ellipses.

5.1 Superimposed Signals

Consider a vector observation of superimposed signals in i.i.d. Gaussian noise:

$$\mathbf{y} = \sum_{k=1}^K \mathbf{s}_k(\mathbf{x}_k) + \boldsymbol{\varepsilon}, \quad (5.1)$$

where the collection of unknowns is defined by:

$$\mathbf{x} \triangleq [\mathbf{x}'_1, \dots, \mathbf{x}'_K]'$$

Each $\mathbf{s}_k(\mathbf{x}_k)$ is a vector of L samples of a signal that is a known function of the unknown parameters \mathbf{x}_k , which we define to be vectors of length M . For the ellipse estimation problem illustrated in Figure 5.1, each \mathbf{x}_k corresponds to the unknown ellipse parameters ($M = 5 + P$), and $\mathbf{s}_k(\mathbf{x}_k)$ corresponds to the discrete sampled projections ($L = PW$), as defined in Chapter 2.

Our goal is to compute the maximum likelihood (ML) estimate of \mathbf{x} given \mathbf{y} :

$$\hat{\mathbf{x}}_{\text{ML}} = \arg \max_{\mathbf{x}} f_{\mathbf{Y}}(\mathbf{y}; \mathbf{x}) = \arg \min_{\mathbf{x}} \left\| \mathbf{y} - \sum_{k=1}^K \mathbf{s}_k(\mathbf{x}_k) \right\|^2, \quad (5.2)$$

where the second equality follows from assuming the measurement error $\boldsymbol{\varepsilon}$ is AWGN. If the number of signals is large (2 or more), brute force minimization of (5.2) would require a global search over many parameters, which is computationally impractical. The EM and AM algorithms are iterative methods for estimating the parameters.

Throughout this chapter, we assume the number of signals K is known. If the number is unknown, then it must be estimated as well. However, the ML criterion is inappropriate for estimating K , since adding parameters will in general increase the “likelihood” (5.2). The MDL and AIC criteria [117, 118, 119, 120, 121] are more suitable candidates for the problem of estimating the number of signals.

5.2 Estimate-Maximize Algorithm

The difficulty in multiple signal estimation problems lies in the reduction of information due to the superposition of the signals. If each signal were observed in isolation, then the parameter estimation problem (5.2) would decompose into K separate minimization problems. For ellipse reconstruction, these minimizations would correspond to matched filters (for position estimates, with a bank of matched filters for shape parameters). The EM algorithm is suited to problems with this characteristic reduction in dimension [111].

We begin by presenting the EM algorithm in its most general setting. Suppose that the measurements \mathbf{y} are a realization of a random vector \mathbf{Y} which has a known distribution $f_{\mathbf{Y}}(\mathbf{y}; \mathbf{x})$. Furthermore, suppose that the measurements can be expressed as the following observation:

$$\mathbf{Y} = \mathbf{HZ} \quad (5.3)$$

where \mathbf{Z} is a random vector representing the “complete data,” whose distribution depends on the unknown parameters \mathbf{x} . We assume \mathbf{H} has more columns than rows, so that the mapping from \mathbf{Z} to \mathbf{Y} is many to one. If we define

$$U(\mathbf{x}; \mathbf{x}_o) = E \{ \log f_{\mathbf{Z}}(\mathbf{z}; \mathbf{x}) | \mathbf{Y} = \mathbf{y}; \mathbf{x}_o \},$$

then the EM algorithm for parameter estimation can be expressed by the following iteration:

E-step:

$$\text{Compute } U(\mathbf{x}; \hat{\mathbf{x}}^{(i)}),$$

M-step:

$$\hat{\mathbf{x}}^{(i+1)} = \arg \max_{\mathbf{x}} U(\mathbf{x}; \hat{\mathbf{x}}^{(i)}),$$

where $\hat{\mathbf{x}}^{(i)}$ denotes the parameter estimate after the i^{th} iteration. The basic idea is to compute U , the conditional expectation of the complete data given the most recent parameter estimate, and then to maximize the parameter’s likelihood as though the complete data were actually observed [111].

Under certain regularity conditions, the EM algorithm is known to be monotone in likelihood. Since it is bounded by the maximum likelihood, it will converge [111, 122]. If the algorithm is initialized sufficiently close (in parameter space) to the ML estimates, then:

$$\hat{\mathbf{x}}^{(i)} \rightarrow \hat{\mathbf{x}}_{\text{ML}}.$$

The distinction between the E-step and the M-step is somewhat artificial in this general setting, but is more clear in the important Gaussian case. If the complete data are normally distributed with mean $\boldsymbol{\mu}(\mathbf{x})$ and covariance $\boldsymbol{\Lambda}$:

$$\mathbf{Z} \sim N(\boldsymbol{\mu}(\mathbf{x}), \boldsymbol{\Lambda}),$$

then it follows from the appendix of [112] that:

$$\begin{aligned} U(\mathbf{x}; \mathbf{x}_o) &= \\ &= c_1 - \frac{1}{2} \left(\boldsymbol{\mu}(\mathbf{x})' \boldsymbol{\Lambda}^{-1} \boldsymbol{\mu}(\mathbf{x}) - 2 \boldsymbol{\mu}(\mathbf{x})' \boldsymbol{\Lambda}^{-1} \left[\boldsymbol{\mu}(\mathbf{x}_o) + \boldsymbol{\Lambda} \mathbf{H}' (\mathbf{H} \boldsymbol{\Lambda} \mathbf{H}')^{-1} (\mathbf{y} - \mathbf{H} \boldsymbol{\mu}(\mathbf{x}_o)) \right] \right) \\ &= c_2 - \left\| \boldsymbol{\Lambda}^{\frac{1}{2}} \left(\boldsymbol{\mu}(\mathbf{x}) - \left[\boldsymbol{\mu}(\mathbf{x}_o) + \boldsymbol{\Lambda} \mathbf{H}' (\mathbf{H} \boldsymbol{\Lambda} \mathbf{H}')^{-1} (\mathbf{y} - \mathbf{H} \boldsymbol{\mu}(\mathbf{x}_o)) \right] \right) \right\|^2, \end{aligned}$$

where c_1 and c_2 are independent of \mathbf{x} . In this case, the EM iterations can be rewritten:

E-step:

$$\hat{\boldsymbol{\mu}}^{(i)} = \boldsymbol{\mu}(\hat{\mathbf{x}}^{(i)}) + \boldsymbol{\Lambda} \mathbf{H}' (\mathbf{H} \boldsymbol{\Lambda} \mathbf{H}')^{-1} (\mathbf{y} - \mathbf{H} \boldsymbol{\mu}(\hat{\mathbf{x}}^{(i)})), \quad (5.4)$$

M-step:

$$\hat{\mathbf{x}}^{(i+1)} = \arg \min_{\mathbf{x}} \left\| \boldsymbol{\Lambda}^{\frac{1}{2}} \left(\hat{\boldsymbol{\mu}}^{(i)} - \boldsymbol{\mu}(\mathbf{x}) \right) \right\|^2. \quad (5.5)$$

In this case, the E-step consists of estimating the mean of the complete data, and the M-step attempts to find the parameters that agree most closely with that estimate.

In general, there is no guarantee that the EM algorithm will be any more tractable computationally than a brute force search or an iterative descent algorithm. The key lies in the selection of the complete data \mathbf{Z} . As mentioned above, the parameter estimation would be relatively straightforward if the signals were observed separately. Therefore, a natural choice for the complete data is the concatenation of the individual signals. Since (5.3) does not contain an additive noise term, we must also distribute the measurement error $\boldsymbol{\varepsilon}$ within the complete data. Therefore, we define:

$$\mathbf{z} = \begin{bmatrix} \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_K \end{bmatrix} = \boldsymbol{\mu}(\mathbf{x}) + \mathbf{n} = \begin{bmatrix} \mathbf{s}_1(\mathbf{x}_1) \\ \vdots \\ \mathbf{s}_K(\mathbf{x}_K) \end{bmatrix} + \begin{bmatrix} \mathbf{n}_1 \\ \vdots \\ \mathbf{n}_K \end{bmatrix},$$

where

$$\begin{aligned} \mathbf{n}_k &\sim N(\mathbf{0}, \beta_k \sigma^2 \mathbf{I}), \\ \sum_{k=1}^K \beta_k &= 1, \beta_k \geq 0. \end{aligned}$$

The constants β_k are arbitrary, and are generally set to $1/K$ or to the normalized SNR [112]:

$$\beta_k = \frac{\|\mathbf{s}_k(\hat{\mathbf{x}}_k)\|^2}{\sum_{j=1}^K \|\mathbf{s}_j(\hat{\mathbf{x}}_j)\|^2}$$

Since $\sum_{k=1}^K \mathbf{n}_k$ is equal (in law) to $\boldsymbol{\varepsilon}$, the original measurement model (5.1) can be rewritten in terms of the EM measurement model (5.3) by defining:

$$\mathbf{y} = \mathbf{H}\mathbf{z}$$

where

$$\begin{aligned} \mathbf{H} &= \mathbf{1}' \otimes \mathbf{I}_L, \\ \mathbf{1} &= \underbrace{[1 \cdots 1]'}_{K \text{ terms}}, \end{aligned}$$

and \otimes denotes the Kronecker product. The covariance of the complete data is then given by:

$$\boldsymbol{\Lambda} = \mathbf{D}(\boldsymbol{\beta}) \otimes \mathbf{I}_L,$$

where

$$\boldsymbol{\beta} \triangleq [\beta_1, \dots, \beta_K]', \quad \mathbf{D}(\boldsymbol{\beta}) \triangleq \text{diag}(\boldsymbol{\beta}).$$

After substituting these definitions into (5.4), simple matrix algebra yields:

$$\hat{\boldsymbol{\mu}}^{(i)} = \boldsymbol{\mu}(\hat{\mathbf{x}}^{(i)}) + (\boldsymbol{\beta} \otimes \mathbf{I}_L)(\mathbf{y} - \mathbf{H}\boldsymbol{\mu}(\hat{\mathbf{x}}^{(i)})).$$

Substituting this expression into (5.5) yields the following iterative algorithm for estimating the parameters of superimposed signals:

E-step:

$$\hat{\boldsymbol{\epsilon}}^{(i)} = \mathbf{y} - \sum_{k=1}^K \mathbf{s}_k(\hat{\mathbf{x}}_k^{(i)}), \quad (5.6)$$

$$\begin{aligned} \hat{\mathbf{s}}_{\text{EMref},k}^{(n)} &= \beta_k \hat{\boldsymbol{\epsilon}}^{(i)} + \mathbf{s}_k(\hat{\mathbf{x}}_k^{(i)}), \quad k = 1, \dots, K, \\ &= \beta_k \left(\mathbf{y} - \sum_{j=1, j \neq k}^K \mathbf{s}_j(\hat{\mathbf{x}}_j^{(i)}) \right) + (1 - \beta_k) \mathbf{s}_k(\hat{\mathbf{x}}_k^{(i)}), \end{aligned} \quad (5.7)$$

M-step: for $k = 1, \dots, K$:

$$\hat{\mathbf{x}}_k^{(i+1)} = \arg \min_{\mathbf{x}_k} \|\hat{\mathbf{s}}_{\text{EMref},k}^{(n)} - \mathbf{s}_k(\mathbf{x}_k)\|^2.$$

The E-step computes $\hat{\mathbf{s}}_{\text{EMref},k}^{(n)}$, the reference estimate of the k^{th} signal based on the current parameter estimate. The M-step then finds a new set of parameters whose signals agree most closely with the reference signals. We now see the potential for computational savings, since the minimizations within the M-step can be performed independently. In fact, this algorithm is very amenable to a parallel implementation since most of the computational cost is in the M-step. Unfortunately, this gain may be negated by the fact that the EM algorithm may require many iterations for convergence [111].

5.3 Alternate-Maximize Algorithm

A very simple approach to nonlinear minimization is to minimize iteratively the objective function over each unknown, while holding the other unknowns fixed. For the superimposed signals problem, it is more natural to group together the parameters corresponding to each signal, i.e., to estimate each signal's parameters iteratively while holding the other signals fixed. This leads to the following iterative algorithm for computing the ML estimates:

for $i = 0$ to convergence

 for $k = 1, \dots, K$

$$\hat{\mathbf{x}}_k^{(i+1)} = \arg \min_{\mathbf{x}_k} \left\| \left[\mathbf{y} - \sum_{j=1}^{k-1} \mathbf{s}_j(\hat{\mathbf{x}}_j^{(i+1)}) - \sum_{j=k+1}^K \mathbf{s}_j(\hat{\mathbf{x}}_j^{(i)}) \right] - \mathbf{s}_k(\mathbf{x}_k) \right\|^2$$

 end

end.

Again, since we are increasing the likelihood at each iteration, and since the likelihood is bounded above by the ML estimate, this sequence of parameter estimates is guaranteed to converge.

To compare the EM and the AM algorithms directly, we first express the innermost loop of the AM algorithm in a two-step form analogous to the EM algorithm:

for $k = 1, \dots, K$:

E step:

$$\hat{\mathbf{e}}^{(i,k)} = \mathbf{y} - \sum_{j=1}^{k-1} \mathbf{s}_j(\hat{\mathbf{x}}_j^{(i+1)}) - \sum_{j=k}^K \mathbf{s}_j(\hat{\mathbf{x}}_j^{(i)}), \quad (5.8)$$

$$\begin{aligned} \hat{\mathbf{s}}_{\text{AMref},k}^{(n)} &= \hat{\mathbf{e}}^{(i,k)} + \mathbf{s}_k(\hat{\mathbf{x}}_k^{(i)}) \\ &= \mathbf{y} - \sum_{j=1}^{k-1} \mathbf{s}_j(\hat{\mathbf{x}}_j^{(i+1)}) - \sum_{j=k+1}^K \mathbf{s}_j(\hat{\mathbf{x}}_j^{(i)}), \end{aligned} \quad (5.9)$$

M step:

$$\hat{\mathbf{x}}_k^{(i+1)} = \arg \min_{\mathbf{x}_k} \left\| \hat{\mathbf{s}}_{\text{AMref},k}^{(n)} - \mathbf{s}_k(\mathbf{x}_k) \right\|^2 \quad (5.10)$$

end.

There are two important distinctions between the two algorithms. First, the timing of the updates is different because the reference signal $\hat{\mathbf{s}}_{\text{AMref},k}^{(n)}$, and hence the estimate $\hat{\mathbf{x}}_k^{(i+1)}$, depends on $\hat{\mathbf{x}}_1^{(i+1)}, \dots, \hat{\mathbf{x}}_{k-1}^{(i+1)}$. This is due to the fact that the residual estimate $\hat{\mathbf{e}}^{(i,k)}$ is updated at every parameter update (compare (5.8) and (5.6)). Therefore, the AM algorithm is not amenable to a parallel implementation. However, it may converge in fewer iterations in a serial implementation since the most recent estimates are used to form the reference signal (5.9). Second, the reference signals themselves are formed rather differently (compare (5.7) and (5.9)). Both the EM and the AM algorithm compute an estimate of the residual $\hat{\mathbf{e}}^{(i)}$, but the EM algorithm adds only a fraction β_k of that residual to the previous signal estimate $\mathbf{s}_k(\hat{\mathbf{x}}_k^{(i)})$ to form the reference signal. Note that the residual signal is important because it contains any remaining signal error as well as the measurement noise. Thus, the EM algorithm is more cautious about straying away from the initial estimate, which may explain why it can be slow to converge. On the other hand, we see from the second equality in (5.9) that the AM reference is formed with total disregard for $\mathbf{s}_k(\hat{\mathbf{x}}_k^{(i)})$, which makes it a greedier algorithm.

The importance of caution depends on how much confidence one has in the initial estimates. In the next section we discuss one simple method for obtaining the initial estimates. Based on our empirical work with ellipse reconstruction, the caution of the EM algorithm was unjustified and only delayed the convergence. A theoretical illustration of the slow convergence of the EM algorithm is given in Section 5.8.

5.4 Initialization

Both of the above iterative algorithms require an initial estimate for the parameters $\mathbf{x}_1, \dots, \mathbf{x}_K$. We found the following variation on the approach of Kwakernaak [123] useful for the multiple ellipse estimation problem [83, 82]. First, estimate the parameters as though there were only one ellipse. Second, subtract the projection of the estimated ellipse (i.e. the estimated signal) from the measurements. Then repeat those steps until K signals have been estimated suboptimally. This initialization procedure is actually equivalent to the first K steps of the AM algorithm with all of the initial parameters set to zero. We found empirically that this approach worked well, provided that the projections of each ellipse were overlap free in at least two of the P projections. When there was overlap in all but one projection, then the initial estimate was often sufficiently far from the ML parameters that the subsequent iterations by the AM or EM algorithms led only to local extrema.

5.5 Number of Views

Any method for reconstruction from projections should be accompanied by a theorem that ensures the uniqueness of the reconstruction. In conventional computerized tomography, the projection-slice theorem provides the necessary theoretical justification [124]. However, if we have only a small number of projections, then the projection-slice theorem is insufficient.

Bresler [15] addressed the problem of determining how many projection angles are required to determine a set of ellipses uniquely. The two main results of his analysis are as follows:

Theorem 5.1 (Thm. 2.12 of [15]) *Any set of K disjoint ellipses can be reconstructed uniquely from any set of its $K+2$ distinct continuous projections.*

Theorem 5.2 (Thm. 2.13 of [15]) *The reconstruction of any random K -ellipse distribution from its projections at any three [different] view-angles is unique with probability one.*

These theorems guarantee sufficiency provided the projections are sampled at an interval smaller than $1/4$ the minimum of the lengths of the minor axes of the ellipses [15]. In fact, if the sampling interval is greater than $3/8$ the minimum minor axis, then unique reconstruction cannot be guaranteed [15]. Furthermore, these theorems assume that the projections are noiseless. In the presence of measurement error and blurring, we conjecture that four views is the minimum plausible number.

5.6 Optimal Projection Angles

Although Theorem 5.2 guarantees the uniqueness of reconstructing from only three views, it does not provide any insight for selecting projection angles in the presence of noise. In this section, we present an analysis of the error in estimating the ellipse position, which suggests an optimal set of view angles. At first this approach may seem inappropriate, since an ellipse's shape parameters are often more important diagnostically than its position parameters. However, we have found empirically that estimates of the shape parameters are rather sensitive to errors in the position parameters. Fortunately, the position can usually be estimated quite accurately by incorporating arterial smoothness.

For simplicity, we consider only a single ellipse, since the optimal projection angles for a set of ellipses would depend on the ellipse locations in general. Thus, consider an ellipse with parameters $\mathbf{x} = [c_x \ c_y \ r \ \lambda \ \phi \ \rho]'$, and with projections

$$\mathbf{y}_p = \mathbf{s}_{\theta_p}(\mathbf{x}) + \boldsymbol{\varepsilon}_p,$$

for $p = 1, \dots, P$. The ML estimate of the ellipse parameters is given by:

$$\hat{\mathbf{x}}_{\text{ML}} = \arg \min_{\mathbf{x}} \sum_{p=1}^P \|\mathbf{y}_p - \mathbf{s}_{\theta_p}(\mathbf{x})\|^2.$$

The most natural approach to finding the optimal projection angles would be to find the angles that minimize the MSE:

$$\hat{\theta}_1, \dots, \hat{\theta}_P = \arg \min_{\theta_1, \dots, \theta_P} E \{ \|\hat{\mathbf{x}}_{\text{ML}} - \mathbf{x}\|^2 \},$$

where the latter expectation should be taken with respect to a random distribution for \mathbf{x} to remove any dependence on ϕ . This criterion can be evaluated numerically through Monte Carlo simulation, but we gain more insight by considering the following approximate analysis.

Recall from Figure 3.5 that the projection \mathbf{s}_{θ_p} of an ellipse is a semi-ellipse function whose position depends on the ellipse position. From (3.4), the relationship is given by:

$$u_{\theta_p} \triangleq c_x \sin \theta_p - c_y \cos \theta_p,$$

where u_{θ_p} denotes the position in the projection. If the ellipse radius is known approximately, then a simple matched filter can be used to estimate u_{θ_p} independently in each of the P projections. Due to noise, there will be some error δ_p in this estimation, i.e.:

$$\hat{u}_{\theta_p} = u_{\theta_p} + \delta_p.$$

By the symmetry of a semi-ellipse, the estimation error δ_p will be unbiased. The P estimates can be aggregated as follows:

$$\hat{\mathbf{u}} \triangleq \mathbf{A}(\boldsymbol{\theta})\mathbf{c} + \boldsymbol{\delta},$$

where

$$\begin{aligned}\mathbf{u} &\triangleq [u_{\theta_1}, \dots, u_{\theta_P}]', \\ \mathbf{c} &\triangleq [c_x \ c_y]', \\ \boldsymbol{\delta} &\triangleq [\delta_1, \dots, \delta_P]', \\ \boldsymbol{\theta} &\triangleq [\theta_1, \dots, \theta_P]'\end{aligned}$$

and

$$\mathbf{A}(\boldsymbol{\theta}) \triangleq \begin{bmatrix} \sin \theta_1 & -\cos \theta_1 \\ & \vdots \\ \sin \theta_P & -\cos \theta_P \end{bmatrix}.$$

We can obtain an estimate for \mathbf{c} from $\hat{\mathbf{u}}$ by linear least-squares:

$$\hat{\mathbf{c}} = (\mathbf{A}(\boldsymbol{\theta})' \mathbf{A}(\boldsymbol{\theta}))^{-1} \mathbf{A}(\boldsymbol{\theta})' \hat{\mathbf{u}}.$$

To find the optimal projection angles, we consider the error covariance of $\hat{\mathbf{c}}$:

$$\begin{aligned}\mathbf{C}_{\boldsymbol{\theta}} &\triangleq \text{Cov}\{\hat{\mathbf{c}} - \mathbf{c}\} \\ &= E\{(\hat{\mathbf{c}} - \mathbf{c})(\hat{\mathbf{c}} - \mathbf{c})'; \boldsymbol{\theta}\} \\ &= \text{Cov}\{(\mathbf{A}(\boldsymbol{\theta})' \mathbf{A}(\boldsymbol{\theta}))^{-1} \mathbf{A}(\boldsymbol{\theta})' \hat{\mathbf{u}} - \mathbf{c}\} \\ &= \text{Cov}\{(\mathbf{A}(\boldsymbol{\theta})' \mathbf{A}(\boldsymbol{\theta}))^{-1} \mathbf{A}(\boldsymbol{\theta})' (\mathbf{A}(\boldsymbol{\theta}) \mathbf{c} + \boldsymbol{\delta}) - \mathbf{c}\} \\ &= \text{Cov}\{(\mathbf{A}(\boldsymbol{\theta})' \mathbf{A}(\boldsymbol{\theta}))^{-1} \mathbf{A}(\boldsymbol{\theta})' \boldsymbol{\delta}\}.\end{aligned}$$

Since the projections are processed independently, the components of $\boldsymbol{\delta}$ are independent. To make the problem tractable, we assume the components of $\boldsymbol{\delta}$ have equal variance¹:

$$\boldsymbol{\delta} \sim N(\mathbf{0}, \sigma_{\delta}^2 \mathbf{I}).$$

Therefore:

$$\begin{aligned}\mathbf{C}_{\boldsymbol{\theta}} &= (\mathbf{A}(\boldsymbol{\theta})' \mathbf{A}(\boldsymbol{\theta}))^{-1} \mathbf{A}(\boldsymbol{\theta})' E\{\boldsymbol{\delta} \boldsymbol{\delta}'\} \mathbf{A}(\boldsymbol{\theta}) (\mathbf{A}(\boldsymbol{\theta})' \mathbf{A}(\boldsymbol{\theta}))^{-1} \\ &= \sigma_{\delta}^2 (\mathbf{A}(\boldsymbol{\theta})' \mathbf{A}(\boldsymbol{\theta}))^{-1} \\ &= \sigma_{\delta}^2 \begin{bmatrix} \sum_{p=1}^P \sin^2 \theta_p & -\sum_{p=1}^P \sin \theta_p \cos \theta_p \\ -\sum_{p=1}^P \sin \theta_p \cos \theta_p & \sum_{p=1}^P \cos^2 \theta_p \end{bmatrix}^{-1} \\ &= \sigma_{\delta}^2 \frac{\begin{bmatrix} \sum_{p=1}^P \cos^2 \theta_p & \sum_{p=1}^P \sin \theta_p \cos \theta_p \\ \sum_{p=1}^P \sin \theta_p \cos \theta_p & \sum_{p=1}^P \sin^2 \theta_p \end{bmatrix}}{\left(\sum_{p=1}^P \cos^2 \theta_p\right) \left(\sum_{p=1}^P \sin^2 \theta_p\right) - \left(\sum_{p=1}^P \sin \theta_p \cos \theta_p\right)^2}.\end{aligned}$$

¹In X-ray imaging, the noise variance will differ between the projections that pass through the long and short axes of the body, so the components of $\boldsymbol{\delta}$ may have different variance.

It is interesting to examine the special case where $P = 2$:

$$\mathbf{C}_{[0,\theta]} = \sigma_\delta^2 \begin{bmatrix} \frac{1+\cos^2\theta}{\sin^2\theta} & \cot\theta \\ \cot\theta & 1 \end{bmatrix}.$$

For θ close to zero, the x error variance grows very large. This is a classic problem with 3-D reconstruction from a pair of stereo angiograms, and with triangulation methods in general.

To minimize the error variance, we would like the diagonal elements of \mathbf{C}_θ to be small. We first introduce two constraints.

Constraint 5.1 *Position-independent error variance.*

For symmetry, we would like the x and y errors to have the same variance, so $\sum_{p=1}^P \sin^2 \theta_p = \sum_{p=1}^P \cos^2 \theta_p$. Note that this constraint implies that:

$$\begin{aligned} \sum_{p=1}^P \sin^2 \theta_p &= \sum_{p=1}^P \cos^2 \theta_p \\ &= \sum_{p=1}^P (1 - \sin^2 \theta_p) \\ &= P - \sum_{p=1}^P \sin^2 \theta_p \\ \Rightarrow \sum_{p=1}^P \sin^2 \theta_p &= \sum_{p=1}^P \cos^2 \theta_p = P/2. \end{aligned}$$

Constraint 5.2 *Uncorrelated position errors.*

For rotational invariance, we would like the x and y position errors to be uncorrelated, which requires $\sum_{p=1}^P \sin \theta_p \cos \theta_p = 0$. Under these two constraints:

$$\mathbf{C}_\theta = \sigma_\delta^2 \begin{bmatrix} \frac{2}{P} & 0 \\ 0 & \frac{2}{P} \end{bmatrix},$$

so there is not much left to minimize! In fact, the problem reduces simply to finding sets of angles that satisfy the two constraints. One such set of angles is:

$$\theta = \left[0, \frac{\pi}{P}, \dots, \frac{(P-1)\pi}{P} \right],$$

i.e., an equally spaced set of angles spanning $[0, \pi]$ is an optimal set.

For $P = 4$, another set of projection angles which satisfies the two constraints is $\theta = \{0^\circ, \theta_r, 90^\circ, 90^\circ + \theta_r\}$, which, for small θ_r , may be more practical for bi-plane X-ray systems than a full 45° rotation. We can now see the inadequacy of this position-only analysis, since, unlike the position error variance \mathbf{C}_θ , the shape

error variance will be dependent on θ_r . Furthermore, this result is incomplete for multiple objects, since the position error variance for two ellipses would also depend on θ_r .

Although this analysis is incomplete, it is comforting that a simple analysis confirms the intuitive choice of an equally spaced set of projection angles. An interesting extension of this result would be to examine the shape error variance as a function of θ_r .

5.7 Conclusion

Though interesting theoretically, we abandoned the hierarchical approach of estimating ellipses on a slice-by-slice basis in favor of the object-based approach of Chapter 6. The hierarchical method was most appropriate for disjoint objects, which limited its applicability to simulations [83, 82]. In particular, an object-based approach is more appropriate for branching objects, since the information that identifies a bifurcation is more global than the information that is available in a single slice.

5.8 Appendix: Slow EM Convergence

As a simple example of how slow the EM convergence can be, we consider the case where the signals are a linear function of the parameters, i.e., $\mathbf{s}_k(\mathbf{x}_k) = \mathbf{A}_k \mathbf{x}_k$, so that:

$$\mathbf{y} = \sum_{k=1}^K \mathbf{A}_k \mathbf{x}_k + \varepsilon = \mathbf{A} \mathbf{x}, \quad (5.11)$$

where

$$\mathbf{A} \triangleq [\mathbf{A}_1, \dots, \mathbf{A}_K].$$

For the ML estimate to be unique, \mathbf{A} must have full column rank.

In this case, the EM iterations can be expressed as the following recursion:

for $k = 1, \dots, K$

$$\hat{\mathbf{x}}_k^{(i+1)} = (\mathbf{A}'_k \mathbf{A})^{-1} \mathbf{A}'_k [\beta_k (\mathbf{y} - \mathbf{A} \hat{\mathbf{x}}^{(i)}) + \mathbf{A}_k \hat{\mathbf{x}}_k^{(i)}]$$

end.

These updates can be aggregated as follows:

$$\begin{aligned} \hat{\mathbf{x}}^{(i+1)} &= \hat{\mathbf{x}}^{(i)} + \mathbf{B} \mathbf{A}' (\mathbf{y} - \mathbf{A} \hat{\mathbf{x}}^{(i)}) \\ &= (\mathbf{I} - \mathbf{B} \mathbf{A}' \mathbf{A} \hat{\mathbf{x}}^{(i)}) + \mathbf{B} \mathbf{A}' \mathbf{y}, \end{aligned} \quad (5.12)$$

where

$$\mathbf{B} \triangleq \begin{bmatrix} \beta_1 (\mathbf{A}'_1 \mathbf{A}_1)^{-1} & & \\ & \ddots & \\ & & \beta_K (\mathbf{A}'_K \mathbf{A}_K)^{-1} \end{bmatrix}.$$

The exact solution for this problem is the standard least-squares estimate:

$$\hat{\mathbf{x}} = (\mathbf{A}' \mathbf{A})^{-1} \mathbf{A}' \mathbf{y}.$$

If we let δ_i denote the error at the i^{th} iteration, i.e.:

$$\hat{\mathbf{x}}^{(i)} = \hat{\mathbf{x}} + \delta_i,$$

then (5.12) can be rewritten:

$$\delta_{i+1} = (\mathbf{I} - \mathbf{B} \mathbf{A}' \mathbf{A}) \delta_i.$$

It follows from this recursive relationship that the rate of convergence is limited by the maximum absolute eigenvalue, or *spectral radius*, of $\mathbf{I} - \mathbf{B} \mathbf{A}' \mathbf{A}$, denoted by $\rho(\mathbf{I} - \mathbf{B} \mathbf{A}' \mathbf{A})$.

Suppose the columns of \mathbf{A} are orthogonal. In this case, the AM algorithm will converge in one iteration (K evaluations of (5.10)). How slowly will the EM algorithm converge? It is easy to show that:

$$\mathbf{I} - \mathbf{B}\mathbf{A}'\mathbf{A} = \begin{bmatrix} (1 - \beta_1)\mathbf{I} & & & \\ & \ddots & & \\ & & \ddots & \\ & & & (1 - \beta_K)\mathbf{I} \end{bmatrix},$$

so that $\rho(\mathbf{I} - \mathbf{B}\mathbf{A}'\mathbf{A}) = \max_k(1 - \beta_k)$. The diagonal factors $(1 - \beta_k)$ can be very close to 1, so the EM algorithm can converge very slowly even in the simple case of orthogonal, linear measurements.

Chapter 6

3-D Object Estimation

Equipped with the measurement and object models of Chapters 2 and 3, we can define criteria for the problem that is the focal point of this dissertation: estimating an unknown object-set Ψ from noisy projection measurements \mathbf{y} . After presenting the maximum-likelihood criterion and noting its shortcomings, we introduce the proposed nonparametric optimality criterion and present a computationally efficient optimization algorithms for both the parallel and the cone-beam projection geometries.

6.1 Maximum Likelihood Criterion

We showed that the single-object ML estimator is inadequate in Chapter 4. It has even more problems in the multiple-object case. Again, the ML estimate is the object-set whose computed projections are the closest to the measurements in the least-squares sense (cf. (3.1)):

$$\hat{\Psi}_{\text{ML}} \triangleq \arg \min_K \min_{\mathcal{Z}_1, \dots, \mathcal{Z}_K} \min_{\mathcal{X}_1, \dots, \mathcal{X}_K} \|\mathbf{y} - \mathbf{s}(\mu_{\Psi})\|^2.$$

This estimator is severely under-regularized; one can always add tiny objects whose parameters fit some bit of noise, thereby increasing the “likelihood,” but certainly not improving the estimate. Even if the number of objects is fixed so that K cannot grow without bound, the results will be unsatisfactory at low SNR since arterial smoothness is not taken into account, as discussed in Chapter 4.

6.2 Nonparametric Optimality Criterion

The nonparametric method of Chapter 4 is, at least conceptually, easy to generalize to the arterial tree reconstruction problem, leading to the following estimator and optimality criterion:

$$\hat{\Psi} \triangleq \arg \min_K \min_{\mathcal{Z}_1, \dots, \mathcal{Z}_K} \min_{\mathbf{x}_1(\cdot), \dots, \mathbf{x}_K(\cdot)} \Phi(\Psi),$$

$$\Phi(\Psi) \triangleq \|\mathbf{y} - \mathbf{s}(\mu_\Psi)\|^2 + \alpha' \sum_{k=1}^K \int_{\mathcal{Z}_k} \ddot{\mathbf{x}}_k^2(z) dz, \quad (6.1)$$

where $\ddot{\mathbf{x}}(z)$ is the $(5 + P)$ -dimensional vector containing the second partial derivatives of the components of $\mathbf{x}(z)$ with respect to z .

Again, the smoothing factor $\alpha = [\alpha_1, \dots, \alpha_{5+P}]'$ controls the tradeoff between fit to the measurements and smoothness of the estimated objects. Intuitively, we would use the same fairly large value for α_1 and α_2 since ellipse position varies relatively slowly, very large identical values for $\alpha_6, \dots, \alpha_{5+P}$, since density typically varies slowly, and smaller values for α_3, α_4 , and α_5 to avoid oversmoothing the important shape features. Again, cross-validation can be used to estimate α from the data itself.

Why this criterion? The first term of (6.1) is the measurement negative-log-likelihood, which we would like to be small, but not at the expense of excessive object wiggleness. Functions that minimize, subject to specified constraints, the second term turn out to be splines, the smoothest functions (in curvature sense) satisfying those constraints. We argue that for maximum effectiveness, arteries attempt to be as smooth as possible, subject to anatomic constraints. An example is carotid arteries, which make very smooth trajectories from their origins at the aorta to their destinations in the head. These arguments are heuristic, and the reader may disagree; we hope this discussion prompts proposals of other criteria. Our main point here is that (6.1) is a natural optimality criterion that encompasses the entire arterial tree globally, and therefore promises better results than local methods.

At least in theory, the optimality criterion (6.1) could be used to solve all of the unsolved problems discussed in Chapter 10. By using different projection functions for $\mathbf{s}(\mu_\Psi)$, arbitrary projection geometries could be considered, by interpreting z as the distance along an artery's medial axis, multi-valued objects could be considered, and by using the outer two minimizations of (6.1), the method could be automated. The practical difficulty lies in finding a computationally efficient algorithm for computing $\hat{\Psi}$, and perhaps more importantly, ensuring that such an algorithm is numerically stable. We restrict our attention here to a semi-automatic method for estimating SGC objects from a parallel geometry.

The particular algorithm for minimizing (6.1) we describe here relies heavily on the fact that, for a parallel (or cylindrical) imaging geometry, the projections of an elliptical cross-section fall on the same row in each of the projection images. This fact is embodied in (2.1) and (2.4), which allows us to rewrite (6.1) as:

$$\hat{\Psi} = \arg \min_K \min_{\mathcal{Z}_1, \dots, \mathcal{Z}_K} \min_{\mathbf{x}_1(\cdot), \dots, \mathbf{x}_K(\cdot)} \left[\sum_{n=1}^N \|\mathbf{y}_n - \mathbf{s}(\mathbf{x}_1(z_n), \dots, \mathbf{x}_K(z_n))\|^2 + \alpha' \sum_{k=1}^K \int \ddot{\mathbf{x}}_k^2(z) dz \right], \quad (6.2)$$

where (cf. (2.4)) the 2-D projection of an ellipse collection is given by:

$$\mathbf{s}(\mathbf{x}_1(z_n), \dots, \mathbf{x}_K(z_n)) = \mathbf{s}_n(\mu_{\{\mathcal{E}(\mathbf{x}_1(z_n)), \dots, \mathcal{E}(\mathbf{x}_K(z_n))\}}). \quad (6.3)$$

By standard arguments [125, 126] based on the Euler equations for the functional (6.2), the infimum of Φ is achieved, and any object-set that achieves that infimum is composed of objects whose component functions are cubic-splines with knots at some subset of z_1, \dots, z_N . A cubic-spline function is determined completely by its values at the knots (sample points). We use this fact to simplify the continuous variational problem (6.2) into a tractable discrete problem. Note that this discretization is a natural consequence of the form of (6.2).

Define the sample points for the k^{th} object by:

$$\mathcal{N}_k \triangleq \{n : z_n \in \mathcal{Z}_k, n = 1, \dots, N\},$$

$$n_{b,k} = \min \mathcal{N}_k, n_{e,k} = \max \mathcal{N}_k,$$

and the samples by:

$$\mathbf{X}_k \triangleq [\mathbf{x}_k(z_{n_{b,k}})', \dots, \mathbf{x}_k(z_{n_{e,k}})']',$$

where \mathbf{X}_k denotes the parameters of the k^{th} object on those planes within its length. With these definitions, we can rewrite (6.2) as:

$$\hat{\Psi} = \arg \min_K \min_{\mathcal{N}_1, \dots, \mathcal{N}_K} \min_{\mathbf{X}_1, \dots, \mathbf{X}_K} \left[\sum_{n=1}^N \|\mathbf{y}_n - \mathbf{s}(\mathbf{x}_1(z_n), \dots, \mathbf{x}_K(z_n))\|^2 + \sum_{k=1}^K \mathbf{X}_k' \mathbf{S}_k \mathbf{X}_k \right]. \quad (6.4)$$

The matrix \mathbf{S}_k , defined by (8.8), depends on \mathcal{N}_k and $\boldsymbol{\alpha}$, and serves to discretize the integral in (6.2).

Though many desirable properties of spline smoothers are known [108, 127], the nonlinearity of (6.4) limits how much we can say about its theoretical properties. There are probably local minima, and even the global minimum is not unique in general, due to the non-uniqueness discussed in Chapter 3. However, regularization methods have shown promise in other applications [109], and the empirical results of Chapter 7 likewise are encouraging.

We have defined an optimality criterion for the object reconstruction problem. This criterion can be used to compare suboptimal algorithms, or can be minimized to generate an arterial tree estimate. In the next section, we present an algorithm for such minimizations.

6.3 Parallel-Projection Estimation Algorithm

Having defined optimality criterion (6.4), the object estimation problem becomes simply the problem of designing an algorithm that can compute $\hat{\Psi}$ with a reasonable computational load. A general nonlinear minimization technique would be completely impractical, since there are thousands of parameters to estimate. Fortunately, we can exploit the sparseness of arterial trees and the special matrix structures of vector-spline smoothing to tailor an algorithm to this problem.

The outer two minimizations in (6.4) can be thought of as a detection operation: estimating the number of arteries and their endpoints. In the remainder of this chapter, we focus on the innermost minimization: the problem of estimating the objects' cross-section parameters given the number of objects and an initial estimate.

An initial estimate could come from the output of any of the sub-optimal 3-D reconstruction schemes, but we currently use manual entry. A trained operator determines the number of objects, and then enters coarse centerlines using a technique similar to [128]. After tracing a coarse piecewise-linear approximation of each object's centerline on one view, the operator traces the centerline in the (e.g.) orthogonal view using auxiliary lines, observing the other views to confirm object correspondence. Since there are multiple views, the correspondence problem that confounds two view reconstruction is alleviated. From cubic-spline interpolants of the 2-D centerlines, an initial 3-D skeleton is generated automatically by analytical back-projection. A typical arterial radius for the anatomy of interest is used as the radius for an initially circular cross-section. The result of this procedure is a crude estimate of Ψ that initializes the iterative algorithm presented below.

If the objects were disjoint, and if their projections were overlap free, then the minimization of (6.4) would decompose into K independent minimizations - one for each object. This fact, combined with the sparseness of arterial trees, suggests that the alternating minimization (AM) algorithm [82, 113] is appropriate for this problem. Here, we use the AM algorithm on an object by object basis: the parameters of each object are estimated in turn, while holding the other objects fixed, and the procedure is iterated¹. If there was no overlap, convergence would occur in one iteration, but in general the number of iterations depends on the degree of overlap. That describes the "outer loop" of the algorithm; we now focus on the algorithm for estimating the parameters of a particular object, holding the other objects fixed.

Suppose we are considering the k^{th} object, and let $\hat{\mathbf{X}}_j$ denote the current parameter estimates of the j^{th} object. Then from the inner minimization of (6.4):

$$\hat{\mathbf{X}}_k = \arg \min_{\mathbf{X}_k} \left[\sum_{n=n_b, k}^{n_e, k} \|\mathbf{y}_n - \mathbf{s}(\hat{\mathbf{x}}_1(z_n), \dots, \hat{\mathbf{x}}_{k-1}(z_n), \mathbf{x}_k(z_n), \hat{\mathbf{x}}_{k+1}(z_n), \dots, \hat{\mathbf{x}}_K(z_n))\|^2 + \mathbf{X}'_k \mathbf{S}_k \mathbf{X}_k \right]$$

¹In [82], the AM algorithm was applied on an ellipse by ellipse basis, and the sparseness argument was weaker.

$$= \arg \min_{\mathbf{X}_k} \left[\sum_{n=n_{b,k}}^{n_{e,k}} \|\mathbf{y}_n^* - \mathbf{s}_n^*(\mathbf{x}_k(z_n))\|^2 + \mathbf{X}_k' \mathbf{S}_k \mathbf{X}_k \right], \quad (6.5)$$

where

$$\begin{aligned} \mathbf{X}_k &\triangleq [\mathbf{x}_k(z_{n_{b,k}})', \dots, \mathbf{x}_k(z_{n_{e,k}})']', \\ \mathbf{y}_n^* &\triangleq \mathbf{y}_n - \mathbf{s}(\hat{\mathbf{x}}_1(z_n), \dots, \hat{\mathbf{x}}_{j-1}(z_n), \hat{\mathbf{x}}_{j+1}(z_n), \dots, \hat{\mathbf{x}}_K(z_n)), \end{aligned} \quad (6.6)$$

and

$$\begin{aligned} \mathbf{s}_n^*(\mathbf{x}) &\triangleq \mathbf{s}(\hat{\mathbf{x}}_1(z_n), \dots, \hat{\mathbf{x}}_{j-1}(z_n), \mathbf{x}, \hat{\mathbf{x}}_{j+1}(z_n), \dots, \hat{\mathbf{x}}_K(z_n)) \\ &\quad - \mathbf{s}(\hat{\mathbf{x}}_1(z_n), \dots, \hat{\mathbf{x}}_{j-1}(z_n), \hat{\mathbf{x}}_{j+1}(z_n), \dots, \hat{\mathbf{x}}_K(z_n)). \end{aligned} \quad (6.7)$$

\mathbf{y}_n^* is the difference between the measurements and the projections of the fixed objects. By our restriction to bifurcations, the k^{th} ellipse on slice n is either isolated, or it intersects one other ellipse, with index k_n , say.

Thus,

$$\mathbf{s}_n^*(\mathbf{x}) = \begin{cases} \mathbf{s}(\mathbf{x}), & \mathbf{x}_k \text{ isolated} \\ \mathbf{s}(\mathbf{x}, \mathbf{x}_{k_n}) - \mathbf{s}(\mathbf{x}_{k_n}), & \mathbf{x}_k \text{ and } \mathbf{x}_{k_n} \text{ intersect} \end{cases}, \quad (6.8)$$

where $\mathbf{s}(\mathbf{x}, \mathbf{x}_{k_n})$ is defined by (6.3). Formulas for $\mathbf{s}_n^*(\mathbf{x})$ are given in Section 3.5. Note that if the objects were disjoint then $\mathbf{s}_n^*(\mathbf{x})$ and $\mathbf{s}(\mathbf{x})$ would be equivalent.

We have finally reduced the original optimization problem (6.4) down to the form (6.5), which is precisely the problem addressed by the algorithm of Chapters 8 and 9. That algorithm uses the nonlinear pseudo-measurement function $\mathbf{s}_n^*(\mathbf{x})$ and its derivatives. The computational efficiency of our nonparametric smoothing algorithm, combined with arterial sparseness, results in an efficient object estimation method.

The cross-validation method described in Chapter 9 could be used to select the smoothing parameter α automatically in several different ways for this problem. A computationally efficient alternative to cross-validating the entire object-set is the following approach. First, estimate the objects using an educated guess for the smoothing parameter. Then, subtract the projections of all of the objects except one (e.g. the longest) from the measurements, leaving (approximately) only the selected object's contribution. Next, apply the cross-validation method of [129] to choose α automatically for that single-object data set. Finally, estimate the entire object-set using the smoothing factor chosen by cross-validation. The examples of Chapter 7, were based on this approach. Unfortunately, the cross-validation score may have local minima, and a descent from the initial α may not yield the truly optimal α . A global search for α on a patient-by-patient basis would be too time consuming; a more practical approach would be to search globally for the best α on a training set of images, and then to apply that value of α to subsequent patient studies.

The object reconstruction algorithm is summarized in Table 6.1. The result of this estimation algorithm is a set of parametrically described SGC objects that are converted to GC parameters using Section 3.6. One

```

Obtain initial estimates  $K$  and  $\mathcal{O}_1, \dots, \mathcal{O}_K$ .
repeat {
  for  $k = 1, \dots, K$ 
    for  $n = n_{b,k}, \dots, n_{e,k}$ 
       $\mathbf{y}_n^* = \mathbf{y}_n - \mathbf{s}(\hat{\mathbf{x}}_1(z_n), \dots, \hat{\mathbf{x}}_{k-1}(z_n), \hat{\mathbf{x}}_{k+1}(z_n), \dots, \hat{\mathbf{x}}_K(z_n))$ .
       $\mathbf{s}_n^*(\mathbf{x}_k(z_n)) = \begin{cases} \mathbf{s}(\mathbf{x}_k(z_n)), & \mathbf{x}_k(z_n) \text{ isolated} \\ \mathbf{s}(\mathbf{x}_k(z_n), \mathbf{x}_{k_n}) - \mathbf{s}(\mathbf{x}_{k_n}), & \mathbf{x}_k(z_n) \text{ intersects } \mathbf{x}_{k_n} \end{cases}$ 
      Compute Jacobians of  $\mathbf{s}_n^*(\mathbf{x})$  at  $\mathbf{x}_k(z_n)$ .
    end
    Apply algorithm of Chapter 9 to  $\{\mathbf{y}_n^*\}$ 
    to obtain new estimates for  $\mathbf{x}_k(z_{n_{b,k}}), \dots, \mathbf{x}_k(z_{n_{e,k}})$ .
  end
} until  $\Phi(\Psi)$  decreases insignificantly.

```

Table 6.1: Iterative object estimation algorithm.

can use these parameters in several ways:

- generate 3-D shaded surface displays directly,
- generate synthetic projections at any angle,
- plot cross-sections, and
- graph parameters (especially radius) versus arterial axis to examine percent stenosis.

Each of these uses is demonstrated in Chapter 7. A bonus for this parametric method is that shaded surface displays are particularly easy to generate since the estimated ellipses are essentially surface descriptions.

6.4 Cone-Beam Reconstruction Algorithm

The algorithm described in the previous two sections is an exact method for computing the optimal arterial tree with respect to criterion (6.1) for parallel or cylindrical projections. In Section 2.3 we presented an approximation that allows us to apply the above algorithm to cone-beam projections of objects close to the axis of rotation. In this section, we generalize the reconstruction algorithm to cone-beam projections.

The exactness of the parallel algorithm is due to the fact that the Euler equations for (6.1) specify that the optimal solution is a set of cubic splines. Thus, the problem of representing a continuous artery by a finite number of ellipses is determined automatically by the mathematics. In fact, the locations of the knots (the ellipse samples) are determined by the measurement resolution, with one ellipse per image row. In the cone-beam case, any object that is away from the axis of rotation will project onto different numbers of rows of pixels in different views due to the position-dependent magnification (fewer pixels in those views to which it is closer). Therefore, there is no mathematically determined discretization. This is a common problem in many engineering optimization applications. The obvious solution is simply to impose a discretization using common sense. Since an object that lies on the axis of rotation would project onto the same rows in each view, we propose using one ellipse per image row to discretize the object. If the vertical resolution is Δ_v , and the on-axis magnification is $M(0)$, then we represent objects by a discrete number of ellipses separated by the distance $\Delta_v/M(0)$. We denote the sample points by z_i , where

$$z_i \triangleq i \left(\frac{\Delta_v}{M(0)} \right).$$

We have now made the problem tractable by discretizing it, but the parallel algorithm is still inapplicable since off-axis objects will project onto different numbers of rows. Fortunately, for typical cone-beam geometries, the numbers of rows differ by only a small number. For example, Tables 6.2 and 6.3 list the starting and ending rows of the objects in the X-ray phantom discussed in Chapter 7. In addition to the row indices at four different projection angles, these tables show the row index that results from using a fixed magnification $M(0)$ as discussed in Section 2.3. This is the ‘ideal’ row index that would result from an on-axis object. Many of the objects end on the rotation axis (see Figure 7.20), so their ending rows $\{n_{e,k}\}_{k=1}^K$ are identical. However, even the starting rows for these objects (that begin up to 5cm off axis) agree to within a few pixels. Thus, simple interpolation should be effective, as we describe below.

Consider the reconstruction algorithm of Table 6.1. This algorithm has two key steps: 1) subtract the contributions of all objects but one from the projection measurements (6.6), and 2) estimate the parameters of the remaining object (6.5) (6.7). These two steps must both be modified for cone-beam projections since there is not a one-to-one relationship between object samples and measurement samples.

k	M(0)	18°	72°	108°	138°
0	-6.2	-6.2	-6.2	-6.2	-6.2
1	-6.2	-7.6	-6.7	-5.8	-5.2
2	-10.5	-8.0	-8.6	-10.0	-11.4
3	-10.5	-12.5	-9.9	-8.5	-7.9
4	-6.2	-7.0	-8.7	-8.7	-7.9
5	-6.2	-6.7	-4.7	-4.0	-4.1
6	-6.2	-8.4	-8.4	-7.3	-6.1
7	-6.2	-5.3	-5.9	-6.5	-6.9

Table 6.2: Starting rows ($n_{b,k}$) of objects in CGR phantom (see Figure 7.20).

k	M(0)	18°	72°	108°	138°
0	220.7	220.7	220.7	220.7	220.7
1	90.9	90.9	90.9	90.9	90.9
2	108.7	108.7	108.7	108.7	108.7
3	41.0	41.0	41.0	41.0	41.0
4	146.7	146.7	146.7	146.7	146.7
5	80.5	80.1	80.1	80.3	80.6
6	52.6	52.8	52.9	52.9	52.8
7	38.5	38.6	38.6	38.5	38.4

Table 6.3: Ending rows ($n_{e,k}$) of objects in CGR phantom (see Figure 7.20).

6.4.1 Cone-Beam Projections of a SGC

We first consider the problem of subtracting the contributions of an object from the measurements. By “contribution” we mean the object’s projections, which for a cone-beam geometry were given in Section 2.3 by:

$$l_{\theta}^{\bullet}(u, v; \mu) = \sqrt{1 + \frac{u^2 + v^2}{d^2}} \int \mu \left(s \cos \theta + \frac{u}{M(s)} \sin \theta, s \sin \theta - \frac{u}{M(s)} \cos \theta, \frac{v}{M(s)} \right) ds, \quad (6.9)$$

where

$$M(s) = \frac{d}{d - d_0 + s} = \frac{d}{d - d_0} \cdot \frac{1}{1 + \frac{s}{d - d_0}}.$$

The density for a SGC object is given by:

$$\mu(x, y, z) = \rho(z) \mathbf{1}_{\mathcal{E}(\mathbf{x}(z))}(x, y),$$

where $\mathbf{x}(z) = [c_x(z) \ c_y(z) \ r(z) \ \lambda(z) \ \phi(z) \ \rho'(z)]'$. The coordinates of the medial axis of the object are given by $(c_x(z), c_y(z), z)$. Consider a particular projection angle θ , then for a given z , let (u_z, v_z) denote the

coordinates of the projection of the medial axis. Then it is easy to see from (6.9) that s_z , u_z , and v_z are given by the solutions to the following system of equations:

$$\begin{aligned} c_x(z) &= s_z \cos \theta + \frac{u_z}{M(s_z)} \sin \theta, \\ c_y(z) &= s_z \sin \theta - \frac{u_z}{M(s_z)} \cos \theta, \\ z &= \frac{v_z}{M(s_z)}. \end{aligned}$$

The solution to the system is:

$$\begin{aligned} s_z &= c_x(z) \cos \theta + c_y(z) \sin \theta, \\ u_z &= M_{\theta,z}(c_x(z) \sin \theta - c_y(z) \cos \theta) \\ v_z &= z M_{\theta,z}, \end{aligned}$$

where

$$M_{\theta,z} \triangleq M(c_x(z) \cos \theta + c_y(z) \sin \theta).$$

These formulae are for computing the projection of the medial axis; the key step for computing the projection of the object is recognizing that the position-dependent magnification changes insignificantly over an elliptical cross-section. Therefore, for $u \approx u_z$:

$$l_{\theta}^{\bullet}(u, v_z; \mu) \approx \sqrt{1 + \frac{u^2 + v_z^2}{d^2}} \int \mu \left(s \cos \theta + \frac{u}{M_{\theta,z}} \sin \theta, s \sin \theta - \frac{u}{M_{\theta,z}} \cos \theta, \frac{v_z}{M_{\theta,z}} \right) ds,$$

so that:

$$\begin{aligned} l_{\theta}^{\bullet}(M_{\theta,z}u, M_{\theta,z}v_z; \mu) &\approx \int \mu (s \cos \theta + u \sin \theta, s \sin \theta - u \cos \theta, v_z) ds, \\ &= l_{\theta}(u, v_z; \mu). \end{aligned}$$

Therefore, to compute the cone-beam projection of an ellipse $\mathbf{x}(z_i)$ at angle θ , we must compute the position-dependent magnification factor $M_{\theta,z}$, and then apply a scaled version of the formulae for $l_{\theta}(u, v; \mathbf{x}(z_i))$ that we derived in Chapter 3.

There is one tricky matter here though. We have decided above to discretize μ by a set of equally spaced ellipses at sample points $\{z_i\}$. The projections of the ellipses will lie on image rows v_{z_i} that will be unequally spaced in general. Since we want to subtract the object's projections from the equally spaced measurement samples, we must generate equally spaced samples of its projections. To do this, we once again exploit the arterial smoothness. Given equally spaced image row coordinates $\{v_n\}_{n=1}^N$, we can find the corresponding points along the medial axis by finding the $z(n)$'s that satisfy $z(n)M_{\theta,z(n)} = v_n$. The set $\{z(n)\}_{n=1}^N$ will be unequally spaced for objects that are away from the rotation axis. If $z(n)$ lies between ellipse samples at z_i and z_{i+1} , then simple interpolation of $\mathbf{x}(z_i)$ and $\mathbf{x}(z_{i+1})$ will produce an ellipse parameter vector that we can then project onto row v_n . Note that the mapping from v_n to $z(n)$ depends on the projection angle θ .

6.4.2 Parameter Estimate Updates

The second step of the algorithm that must be modified for cone-beam projections is the parameter estimation. Given measurements $\mathbf{y}_{p,n}^*$ and a previous estimate $[\hat{\mathbf{x}}(z_{i_b,k}), \dots, \hat{\mathbf{x}}(z_{i_e,k})]$, compute new estimates. The complicating factor is that we have $(i_{e,k} - i_{b,k} + 1)$ sets of ellipse parameters to estimate, but a slightly different number of measurements $\mathbf{y}_{p,n}^*$ in each view. For this step, since the ellipse parameters are the most important, it is more convenient to interpolate the measurements from their original equally-spaced samples to (unequally spaced) samples centered at the points where the ellipses project (v_{z_i}). This is in contrast to the discussion in the previous section, where we interpolated the parameters, since the projections needed to be equally spaced.

The image rows were sampled originally at values v_n , where $v_n = n\Delta_v$. By resampling each column of the images at sample points $\{v_{z_i}\}_{i=i_b,k}^{i_e,k}$, we will create a new set of measurements with the same number of samples at each projection angle. The estimation algorithm of Chapter 9 can then be applied to these resampled measurements. A potential problem with this approach is that the interpolation used to resample the measurements will introduce some measurement correlation. Fortunately, as Tables 6.2 and 6.3 demonstrate, very little interpolation is needed, so the induced correlation should be small.

6.5 Summary

We have described a nonparametric optimality criterion for reconstructing an arterial tree from a few projections. We presented an iterative algorithm for optimizing the criterion given an initial estimate that is currently obtained by manual entry. We proposed a generalization of the algorithm for reconstructing from cone-beam projections, based on resampling of the measurements. It is worth emphasizing that this resampling is object dependent, and is made possible by the fact that we estimate one object at a time iteratively. This type of resampling would be inapplicable to voxel-based cone-beam reconstruction methods, since the different voxels would have conflicting resampling requirements. This is another example of the flexibility of object-based reconstruction.

Chapter 7

Experimental Results

In this chapter, we report the results of applying the algorithm of Chapter 6 to four data sets: simulated projection images, MR projections of a branching flow phantom, MR angiograms of a healthy volunteer’s carotid arteries, and X-ray projections of an aluminum phantom. The SNR per view, defined by $2\rho h_p r_p / \sigma$, and the important imaging parameters for these case studies are summarized in Table 7.1.

7.1 Simulation

Figure 7.1 shows four noisy projection images of a simulated arterial tree. These images were generated using the projection model of Chapter 2. Table 7.2 displays the convergence of the algorithm for the smoothing parameter chosen by cross-validating the longest object. Each iteration through all five objects required about 35 seconds on a SUN Sparcserver. As expected, the first few iterations improve the estimates considerably, with little further improvement after the fourth iteration. The RMS errors for ellipse orientation ϕ

	Simulation	MR phantom	MR carotid	X-ray phantom
K	5	2	4	7
W	256	160	128	256
N	256	128	160	145
Δ_h [mm]	1	0.478	0.7	0.83
Δ_v [mm]	1	0.478	0.35	0.83
P	4	4	4	4
σ^2	3	0.0025	0.015	5.7
SNR	2.8–6.7	6.3–40.6	2.8–20.2	11.2–179.7
$\theta_1, \dots, \theta_P$	0,45,90,135	22,67,112,157	0,45,108,143	0,45,90,135

Table 7.1: Imaging parameters for the data sets used to evaluate the reconstruction method.

Iteration	RMS error					
	c_x	c_y	r	λ	ϕ	ρ
0	1.188	0.9704	1.528	0.06325	23.05	0.2044
1	0.2884	0.2345	0.1943	0.1352	38.25	0.03376
2	0.1859	0.1568	0.136	0.07881	34.66	0.01879
3	0.1783	0.1386	0.1133	0.06639	34.95	0.0145
4	0.1606	0.1174	0.1048	0.07071	34.94	0.01396
5	0.1573	0.1312	0.104	0.06435	34.61	0.01451
6	0.1456	0.1365	0.1081	0.06774	35.2	0.0138
7	0.1556	0.1363	0.1055	0.067	35.05	0.01354
8	0.1507	0.1376	0.1031	0.06741	36.76	0.01365

Table 7.2: RMS parameter estimation errors for eight iterations.

Figure 7.1: Simulated angiograms; $\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$.

are misleadingly large; many of the cross-sections are very close to circular, in which case the orientation is irrelevant.

A more meaningful evaluation of the shape estimates can be made from Figure 7.2, Figures 7.3 and 7.4. (The estimates are taken from the fourth iteration hereafter.) Figure 7.2 compares the true and the estimated cross-sections of the main branch over the stenotic portion. The reconstruction has estimated this eccentric stenosis accurately. Similarly, Figures 7.3 and 7.4 compare the true and the estimated cross-sections of two bifurcations. The overlapping ellipses with very different shapes are estimated accurately.

For a more quantitative comparison, Figure 7.5 compares the true and the estimated radii (both in SGC parameters) for this example, and shows the presence of a stenosis quite clearly. These parameters are translated into GC parameters using the formulae of Section 3.6. Figure 7.6 displays the resulting radius estimation errors for the five objects. The larger radius “errors” at the endpoints of some of the objects are artifacts that are explained by object-set ambiguity discussed in Section 3.2: when one ellipse is almost completely inside another ellipse, it contributes very little to the cross-section. Figures 7.3 and 7.4 show that the cross-sections were in fact estimated accurately.

Graphs such as Figure 7.5 are useful for computing percent stenosis, but morphology is best viewed through an interactive 3-D display with cut-planes. Figures 7.7 and 7.8 show two shaded surface displays of the reconstruction, generated using the simplest aspects of the shading method presented in [130].

The results from this data set demonstrate the potential performance of object-based reconstruction, and highlight an important advantage of the parametric approach: despite the low SNR, the estimates are of sub-pixel accuracy. Discrete voxel representations of μ , on the other hand, are typically limited by the measurement resolution.

Figure 7.2: True (above) versus estimated (below) stenosis cross-sections; $n = 160, \dots, 180$.

Figure 7.3: True (above) versus estimated (below) bifurcation cross-sections; $n = 130, \dots, 141$.

Figure 7.4: True (above) versus estimated (below) bifurcation cross-sections; $n = 80, \dots, 94$.

Figure 7.5: True (solid) versus estimated (dashed) object radii.

Figure 7.6: Radius estimation errors for the simulated objects.

Figure 7.7: 3-D surface display of reconstructed simulated arteries; $\theta = 0^\circ$.

Figure 7.8: 3-D surface display of reconstructed simulated arteries; $\theta = 45^\circ$.

Figure 7.9: Simulated angiograms measurements, $\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$.

Figure 7.10: True cross-sections (top) versus one-ellipse estimates (bottom).

7.2 Crescent Cross-Section Simulation

One of the most common criticisms of an approach based on an elliptical model is that “these parametric approaches usually suffer from a high sensitivity to the exactness of the assumptions used for modelling. This can be redhibitory [*sic*] when actual objects, such as physiological ones, offer a geometry too complex and variable to be easily modelled with few parameters” [71]. In this section, we show a simple example of the versatility of the elliptical model that suggests the parametric approach is less sensitive than might be imagined.

Consider Figure 7.9, these simulated projections are identical to those of Figure 7.1, except that a crescent-shaped stenosis has been added to one of the branches. Obviously, a crescent shape is a severe violation of the elliptical assumption. How “sensitive” are the estimates to this crescent shape? Figure 7.10 displays a comparison between the actual crescent-shaped cross-sections and the estimated elliptical cross-sections. The area of the best-fit ellipses shows remarkable agreement to the area of the crescent lumens. The estimated ellipse shape also seems reasonably robust, considering the non-ellipticity of a crescent.

If a physician wished to test whether a particular arterial segment was crescent shaped, then we can augment the ellipse model by using two ellipses, one for the normal lumen and one for the plaque, and then repeat the estimation. Just as the projection of two overlapping ellipses was computed by considering the set union, here we consider the set difference of the two ellipses: $\mathcal{E}(\mathbf{x}_{\text{lumen}}) \setminus \overline{\mathcal{E}(\mathbf{x}_{\text{plaque}})}$. A simple algorithm for estimating the parameters of the two sets of ellipses is as follows. 1) Identify (by hand) the artery containing a potentially crescent-shaped stenosis, and indicate the extent of the stenosis. 2) Estimate the remainder of the arterial tree using the algorithm of Chapter 6 and interpolate the normal section of the identified artery. 3) Subtract the projections of the estimated arterial tree from the measurements, leaving (the negative of) the projection of the plaque ellipses. 4) Estimate the plaque ellipses from the residual measurements using the algorithm of Chapter 4.

This method was applied to the images in Figure 7.9, and the resulting ellipse estimates are displayed in Figure 7.11. The crescent shape is clearly displayed by the estimates. Note that we have not enforced the constraint that the plaque ellipse should touch the lumen ellipse. Enforcing this constraint would significantly improve the accuracy since the number of degrees of freedom would be reduced.

Figure 7.11: True cross-sections (top) versus two-ellipse estimates (bottom).

Figure 7.12: MR tube-phantom projections; $\theta = 22^\circ, 67^\circ, 112^\circ, 157^\circ$.

Figure 7.13: Histogram of radius estimation error.

7.3 MR Phantom Data

The images in Figure 7.12 are MR projections of a MnCl solution flowing through a plastic Y-shaped connector attached to Tygon tubing. We used the projection-reconstruction selective inversion-recovery (PR SIR) [95, 96] method to collect the data. The inner diameter of the Y connector was 3.75mm. Figure 7.13 displays a histogram of the radius estimation error over the Y section. The RMS error in radius was only 0.04 mm. As a verification of the geometric consistency of the estimates, Figure 7.14 displays the outline of the projection of the estimates superimposed on another view that was also acquired, but was not used for the reconstruction. The overall correspondence is quite good, though the sharp transitions between the small branch and the larger tubes would be better modeled by a GC than a SGC, since some of the image rows intersect both the small tube and a corner of the larger tube. The shaded-surface displays of the estimated objects shown in Figures 7.15 and 7.16 are remarkably similar to the physical phantom.

Figure 7.14: Reprojection of estimates (solid line) on an unused view; $\theta = 0^\circ$.

Figure 7.15: 3-D surface display of reconstructed tube phantom; $\theta = 45^\circ$.

Figure 7.16: 3-D surface display of reconstructed tube phantom; $\theta = -45^\circ$.

7.4 MR Carotid Arteries

Figure 7.17 shows four MRSIR projections of a slab containing a healthy volunteer's right carotid artery. A surface coil was placed on the right side of the subject's neck, to maximize SNR. We have no means of making a quantitative evaluation of the results, but Figures 7.18 and 7.19 show 3-D displays of the estimated arteries. This example clearly motivates extending this work beyond single-valued objects, since this facial artery is clearly a multi-valued function of z .

Figure 7.17: MR *in vivo* carotid angiograms; $\theta = 0^\circ, 45^\circ, 108^\circ, 143^\circ$.

Figure 7.18: 3-D surface display of reconstructed carotid arteries; $\theta = 143^\circ$.

Figure 7.19: 3-D surface display of reconstructed carotid arteries; $\theta = -50^\circ$.

7.5 X-ray Phantom Data

Figure 7.20 shows four X-ray projections of an aluminum phantom built by GE-CGR, acquired on a digital fluoroscopic system. Rougee *et al.* have demonstrated their ART-based reconstruction method on this phantom [71]. These projections have been corrected for the image intensifier's pincushion distortion. The phantom was placed on a rotating turntable for the acquisitions, so the axis of rotation is parallel to the image columns, as we assume.

This data set consists of cone-beam projections, but the objects were close enough to the axis of rotation that we felt it would be worth attempting to use the parallel reconstruction algorithm. As mentioned in Chapter 2, we first applied a local estimation algorithm similar to that of [12] to estimate the PSF on a few of the overlap-free objects, assuming a Gaussian PSF. The estimated σ for the Gaussian PSF was near 0.5 consistently, which corresponds to the following impulse response: $[0.15 \ 0.7 \ 0.15]$. This PSF was held fixed for the 3-D reconstruction.

One of the objects in this phantom is completely horizontal. Since our method does not accommodate multi-valued objects, the image rows containing the horizontal portion were ignored. Figure 7.21 displays a histogram of the radius estimation error. Although most of the errors are below a pixel, there are a few significant errors. These should be corrected by applying the cone-beam reconstruction algorithm. Figures 7.22 and 7.23 show the outline of the reprojections of the estimates superimposed on views that were not used for the estimation.

Figures 7.24 and 7.25 display 3-D shaded displays of the reconstructed objects.

Figure 7.23: Reprojection of estimates (solid line) on an unused view; $\theta = 48^\circ$.

Figure 7.24: Shaded surface of estimate; $\theta = 72^\circ$.

Figure 7.25: Shaded surface of estimate; $\theta = 238^\circ$.

Part II

Nonparametric Smoothing Algorithms

Chapter 8

Linear Nonparametric Smoothing

This chapter presents a novel nonparametric algorithm for smoothing linear vector-valued measurements¹. This algorithm is the core of the nonlinear smoothing algorithm presented in the next chapter, which in turn is the basis of the object-based reconstruction approach described in the previous chapters. However, the development of this algorithm is independent of the preceding chapters, and the algorithm itself is of independent interest, as it provides a nonparametric alternative to the classical Kalman filter.

Spline smoothing has become a popular method for nonparametric exploration and estimation of scalar-valued functions, but its generalizations to vector-valued functions have been underutilized. This chapter presents a computationally efficient algorithm for nonparametric smoothing of vector signals with general measurement covariances. This new algorithm provides an alternative to the prevalent “optimal” smoothing algorithms that hinge on (possibly inaccurate) parametric state-space models. We develop and compare automatic procedures that use the measurements to determine how much to smooth; this adaptation allows the data to “speak for itself” without imposing a Gauss-Markov model structure. We present a nonparametric approach to covariance estimation for the case of i.i.d. measurement errors. Monte Carlo simulations demonstrate the performance of the algorithm.

8.1 Linear Smoothing

The goal of fixed-interval smoothing is to estimate a smooth function from a finite number of noisy measurements. We consider here the linear measurement model:

$$\mathbf{y}_n = \mathbf{H}_n \mathbf{x}_n + \boldsymbol{\varepsilon}_n, \quad n = 1, \dots, N, \quad (8.1)$$

¹This chapter is derived largely from [131].

where

$$\boldsymbol{\varepsilon}_n, \mathbf{y}_n \in \mathfrak{R}^L, \mathbf{x}_n \in \mathfrak{R}^M, \text{ and } \mathbf{H}_n \in \mathfrak{R}^{L \times M}.$$

We assume that the additive measurement error $\boldsymbol{\varepsilon}_n$ is normally distributed with mean zero and (positive definite) covariance matrix $\boldsymbol{\Sigma}_n$, and that the errors are independent between samples. The states $\{\mathbf{x}_n\}$ are (possibly unequally spaced) samples of a process $\mathbf{g}(t_n)$:

$$\mathbf{x}_n = [g_1(t_n), \dots, g_M(t_n)]' \triangleq \mathbf{g}(t_n), t_n < t_{n+1} \forall n,$$

where “ $'$ ” denotes matrix transposition. The goal of smoothing is to estimate \mathbf{g} (and/or its derivatives) from the measurements $\{\mathbf{y}_n\}_{n=1}^N$.

One justifies smoothing by *a priori* knowledge that the component functions of $\mathbf{g}(t)$ vary slowly in some sense. The smoothness of \mathbf{g} is frequently quantified by assuming that the states $\{\mathbf{x}_n\}$ adhere to a parametric Gauss-Markov discrete-time state-space model:

$$\mathbf{x}_{n+1} = \mathbf{A}_n \mathbf{x}_n + \mathbf{B}_n \mathbf{u}_n, \mathbf{u}_n \sim N(\mathbf{0}, \mathbf{Q}_n), \mathbf{x}_0 \sim N(\boldsymbol{\mu}_0, \boldsymbol{\Pi}_0). \quad (8.2)$$

Using such models, one can derive optimal smoothing algorithms [132] that provide minimum mean-square error estimates of the states. However, in many applications the parameters (state evolution matrices and covariances) of the state evolution model (8.2) are unknown, and they must be estimated from the measurements or from a training set [133]. This estimation may result in an inaccurate parametric model.

Rather than impose a possibly inaccurate parametric model, we would sometimes prefer to “let the data speak for itself,” particularly for off-line data exploration. This motivates nonparametric approaches to smoothing [127].

Nonparametric spline smoothing has proven to be successful at estimating scalar-valued functions from noisy data. Therefore, it is not surprising that the nonparametric approach has also been applied in some situations similar to the vector measurement model (8.1). In this chapter, we derive a computationally efficient algorithm for nonparametric smoothing of vector measurements, allowing for general measurement covariances $\boldsymbol{\Sigma}_n$. This has also been considered by Miller and Wegman [134], but their algorithm requires that the covariance matrices be simultaneously diagonalizable. For independent, identically distributed (i.i.d.) measurement errors, we recommend the transformation approach of [134], as it requires fewer computations than the algorithm presented below. Wegman [135], Woltring [136, 137], and Sidhu and Weinert [138] all discuss approaches that assume effectively that the covariance matrices are diagonal. Note that the approach of Sidhu and Weinert [138] does allow for a more general measurement model than (8.1). In the special case of diagonal covariance matrices, the vector-spline smoothing algorithm reduces to repeated applications of the scalar algorithm. However, one can take advantage of any known similarity between the component functions when choosing the smoothing parameters [136, 137], Non-diagonal, non-i.i.d. measurement covariances arise in several problems, including in the nonlinear smoothing algorithm of the next chapter.

Although, as observed by Silverman [108], “non-parametric regression is not as widely known or adopted as perhaps it should be,” spline smoothing concepts have previously had several other generalizations that we list for didactic reasons: estimating a function’s derivatives [108, 139], estimating branching curves [140], smoothing multivariate functions (scalar valued functions of several variables) [141, 127], and estimating curves with discontinuities [142]. Source code for spline smoothing is available² from the `gcv` and `toms` directories of `netlib` [143].

This chapter is organized as follows. In Section 8.2, we review the derivation of the cubic-spline based algorithm for smoothing scalar measurements, following the approach and notation of Reinsch [126, 144]. In Section 8.3, we present the new algorithm for smoothing vector measurements. In Section 8.4, we consider methods that use the measurements to choose automatically the parameters that control the degree of smoothing. In Section 8.5, we describe a nonparametric method for measurement error covariance estimation. In Section 8.6, we outline the algorithm implementation and discuss computational requirements. In Section 8.7, we compare these methods using simulated measurements.

8.2 Spline Smoothing of Scalar Measurements

The Problem

Assume that the scalar measurements y_n satisfy the model

$$y_n = g(t_n) + w_n \varepsilon_n, \quad n = 1, \dots, N,$$

$$\varepsilon_n \sim N(0, \sigma^2), \quad E\{\varepsilon_n \varepsilon_m\} = 0 \text{ if } n \neq m,$$

where $t_1 < \dots < t_N$. The weights w_n are assumed known, but the variance σ^2 may be unknown. Estimation of g by smoothing the y_n ’s always involves a tradeoff between fit to the data and smoothness of the estimated function \hat{g} . For normally distributed measurement errors, the natural measure of fit to the data is the weighted residual sum of squares,

$$\text{RSS}(g) \triangleq \sum_{n=1}^N \left(\frac{y_n - g(t_n)}{w_n} \right)^2.$$

Spline smoothing is based on the following nonparametric measure of the roughness (lack of smoothness) of g :

$$\text{R}_k(g) \triangleq \int_{t_1}^{t_N} (g^{(k)}(t))^2 dt.$$

General differential operators have also been considered, e.g. [138, 134]. For simplicity, we consider here only the case $k = 2$, though the algorithm derived below is fully generalizable. $\text{R}_2(g)$ is related to the

²An e-mail message to `netlib@research.att.com` containing the request ‘send index’ or ‘send index from `gcv`’ will generate a reply containing instructions.

curvature of g , so it weights functions that are very wiggly more heavily. This measure also has the desirable property that $R_2(g) = 0$ if and only if g is linear.

We would like to minimize $RSS(g)$ and $R_2(g)$ simultaneously, but these are conflicting goals in general³. The standard nonparametric solution is to use the curve that minimizes a weighted combination of the two:

$$\hat{g}_\alpha \triangleq \arg \min_g \left[\sum_{n=1}^N \left(\frac{y_n - g(t_n)}{w_n} \right)^2 + \alpha \int (\ddot{g}(t))^2 dt \right]. \quad (8.3)$$

The smoothing parameter α controls the tradeoff between fit to the data and smoothness. As $\alpha \rightarrow 0$, \hat{g}_α approaches the cubic-spline interpolant of the measurements, and as $\alpha \rightarrow \infty$, \hat{g}_α approaches the linear regression of the measurements. Automatic selection of α will be discussed in Section 8.4. $R_2(g)$ acts as a “roughness penalty” [108] that prevents excessive local variation in the curve g . This idea is related to the regularization methods of computer vision [109].

The Solution

As a consequence of the Euler equation corresponding to the variational problem (8.3), the minimizing function \hat{g}_α is a cubic spline [126]. A function g is a cubic spline with knots $\{t_n\}_{n=1}^N$ if and only if there exist coefficients $\{a_n, b_n, c_n, d_n\}_{n=0}^N$ such that:

$$g(t) = a_n + b_n(t - t_n) + \frac{c_n}{2}(t - t_n)^2 + \frac{d_n}{6}(t - t_n)^3, \quad t \in [t_n, t_{n+1}], \quad (8.4)$$

and

$$g(t), \dot{g}(t), \ddot{g}(t) \text{ are all continuous.} \quad (8.5)$$

(t_0 is any number less than t_1 , and t_{N+1} is any number greater than t_N .)

The continuity conditions (8.5) impose a system of equations on the polynomial coefficients that are knot dependent. These equations are [126]:

$$\begin{aligned} b_n &= -\frac{1}{6}h_n(c_{n+1} + 2c_n) + h_n^{-1}(a_{n+1} - a_n), \quad n = 0, \dots, N-1, \\ d_n &= h_n^{-1}(c_{n+1} - c_n), \quad n = 0, \dots, N-1, \\ b_N &= \frac{1}{6}h_{N-1}c_{N-1} + (a_N - a_{N-1})/h_{N-1}, \\ h_n^{-1}a_n - (h_n^{-1} + h_{n+1}^{-1})a_{n+1} + h_{n+1}^{-1}a_{n+2} &= \\ &= \frac{1}{6}h_n c_n + \frac{1}{3}(h_n + h_{n+1})c_{n+1} + \frac{1}{6}h_{n+1}c_{n+1}, \quad n = 0, \dots, N-2, \end{aligned} \quad (8.6)$$

where $h_n = t_n - t_{n-1}$. There are a total of $4(N+1)$ unknowns, so by adding 4 boundary conditions to the above $3N$ equations, we can express all of the coefficients in terms of (a_1, \dots, a_N) . Table 8.1 presents two possible boundary conditions. We restrict our attention here to natural cubic-splines by imposing the

³The solution to the problem of minimizing $R_2(g)$ subject to $g(t_n) = y_n \forall n$ is cubic-spline interpolation. Interpolation is useful only if the measurements are noiseless [126].

Left Boundary	Right Boundary
Natural Splines	
$c_0 = 0$	$c_N = 0$
$d_0 = 0$	$d_N = 0$
Not-a-Knot Splines	
$a_0 = a_1 + b_1 h_1 + \frac{1}{2} c_1 h_1^2 + \frac{1}{6} d_1 h_1^3$	$a_N = a_{N-1} + b_{N-1} h_{N-1} + \frac{1}{2} c_1 h_{N-1}^2 + \frac{1}{6} d_{N-1} h_{N-1}^3$
$d_0 = d_1$	$d_N = d_{N-1}$

Table 8.1: Boundary conditions for spline smoothing.

boundary condition that $\hat{g}_\alpha(t)$ is linear for $t > t_N$ and $t < t_1$. Other boundary conditions, e.g. periodic and complete splines, are possible, which may be important if derivatives of g are to be estimated [139, 145].

For natural cubic-splines, the most important constraints are summarized by the following matrix relation:

$$\mathbf{Q}'\mathbf{a} = \mathbf{T}\mathbf{c}, \quad (8.7)$$

where $\mathbf{a} = (a_1, \dots, a_N)'$, and $\mathbf{c} = (c_2, \dots, c_{N-1})'$. \mathbf{Q} and \mathbf{T} are $N \times (N-2)$ and $(N-2) \times (N-2)$ band matrices respectively:

$$Q_{ji} \triangleq \begin{cases} h_i^{-1}, & j = i \\ -(h_i^{-1} + h_{i+1}^{-1}), & j = i + 1 \\ h_{i+1}^{-1}, & j = i + 2 \\ 0, & \text{otherwise} \end{cases}, \quad T_{ij} \triangleq \begin{cases} \frac{1}{6}h_i, & j = i - 1 \\ \frac{1}{3}(h_i + h_{i+1}), & j = i \\ \frac{1}{6}h_{i+1}, & j = i + 1 \\ 0, & \text{otherwise} \end{cases}.$$

(The B-spline version of \mathbf{Q} and \mathbf{T} is known to result in a numerical algorithm that is more stable [145]; we present this version for simplicity.)

Let $\mathbf{y} = (y_1, \dots, y_N)'$ and $\mathbf{W} = \text{diag}(w_1, \dots, w_N)$. If g is a natural cubic-spline with expansion (8.4), then it was shown in [146] that

$$R_2(g) = \int (\ddot{g}(t))^2 dt = \mathbf{c}'\mathbf{T}\mathbf{c} = \mathbf{a}'\mathbf{Q}\mathbf{T}^{-1}\mathbf{Q}'\mathbf{a}, \quad (8.8)$$

and

$$\text{RSS}(g) = (\mathbf{y} - \mathbf{a})'\mathbf{W}^{-2}(\mathbf{y} - \mathbf{a}).$$

Therefore the coefficients of the smoothing spline \hat{g}_α minimize the quadratic:

$$\hat{\mathbf{a}} = \arg \min_{\mathbf{a}} [(\mathbf{y} - \mathbf{a})'\mathbf{W}^{-2}(\mathbf{y} - \mathbf{a}) + \alpha \mathbf{a}'\mathbf{Q}\mathbf{T}^{-1}\mathbf{Q}'\mathbf{a}].$$

One can find the solution to this minimization by solving the following system of equations for $\hat{\mathbf{a}}$ and \hat{c} [126]:

$$\begin{aligned} \mathbf{Q}'\mathbf{y} &= (\mathbf{T}/\alpha + \mathbf{Q}'\mathbf{W}^2\mathbf{Q})(\hat{c}\alpha) \\ \hat{\mathbf{a}} &= \mathbf{y} - \mathbf{W}^2\mathbf{Q}(\hat{c}\alpha). \end{aligned} \quad (8.9)$$

Since \mathbf{T} and \mathbf{Q} are band matrices, we can solve (8.9) in $O(N)$ operations [147]. These band matrices will also be important to the efficiency of the algorithm for smoothing vector measurements. Having computed $\hat{\mathbf{a}}$ and $\hat{\mathbf{c}}$, we can compute $\hat{\mathbf{b}}$ and $\hat{\mathbf{d}}$ from (8.6), thereby obtaining the piecewise-cubic expansion of \hat{g}_α . In many cases, only $\hat{\mathbf{a}}$ is needed since $\hat{g}_\alpha(t_n) = \hat{a}_n$.

8.3 Spline Smoothing of Vector Measurements

The Problem

We now generalize the results of the previous section by considering the *vector* measurement model⁴:

$$\mathbf{y}_n = \mathbf{g}(t_n) + \boldsymbol{\varepsilon}_n, \quad n = 1, \dots, N, \quad (8.10)$$

$$\mathbf{g}(t_n), \boldsymbol{\varepsilon}_n, \mathbf{y}_n \in \Re^M, \quad \boldsymbol{\varepsilon}_n \sim N(\mathbf{0}, \boldsymbol{\Sigma}_n), \quad E\{\boldsymbol{\varepsilon}_n \boldsymbol{\varepsilon}_m'\} = \mathbf{0}, \quad n \neq m.$$

Although we assume the error covariances $\boldsymbol{\Sigma}_n$ are known for the derivation below, they can be estimated (Section 8.5) if the errors are identically distributed. The goal is to estimate \mathbf{g} from the measurements $\{\mathbf{y}_n\}$.

Again we must compromise between fit to the data and smoothness of the estimated functions. Assuming the errors are normally distributed, the natural measure of fit to the data is

$$\text{RSS}(\mathbf{g}) = \sum_{n=1}^N (\mathbf{y}_n - \mathbf{g}(t_n))' \boldsymbol{\Sigma}_n^{-1} (\mathbf{y}_n - \mathbf{g}(t_n)).$$

Although we assume that the component functions $g_m(t)$ of $\mathbf{g}(t)$ are smooth, they may have different degrees of smoothness, different scales, and different marginal measurement-error variances. Hence, M smoothing parameters, $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_M)$, are required to formulate the problem. However, if a group of the component functions are known to have similar properties, then we equate the corresponding smoothing parameters. The multidimensional generalization of (8.3) is then

$$\hat{\mathbf{g}}\boldsymbol{\alpha} = \arg \min_{\mathbf{g}} \left[\sum_{n=1}^N (\mathbf{y}_n - \mathbf{g}(t_n))' \boldsymbol{\Sigma}_n^{-1} (\mathbf{y}_n - \mathbf{g}(t_n)) + \sum_{m=1}^M \alpha_m \int (\ddot{g}_m(t))^2 dt \right]. \quad (8.11)$$

Before presenting the solution to this general minimization problem, we note two special cases. If the covariances $\boldsymbol{\Sigma}_n$ are diagonal, then (8.11) separates into M independent terms of the form (8.3), so $\hat{\mathbf{g}}\boldsymbol{\alpha}$ can be computed by M evaluations of (8.9). Similarly, if the covariances are simultaneously diagonalizable, then a transformation of (8.11) yields a separable expression [134].

⁴The objection could be raised that model (8.10) is not as general as model (8.1), which contains the additional \mathbf{H}_n term. However, if the measurement matrices \mathbf{H}_n are all of rank M , then multiplying both sides of (8.1) by $(\mathbf{H}_n' \boldsymbol{\Sigma}_n^{-1} \mathbf{H}_n)^{-1} \mathbf{H}_n' \boldsymbol{\Sigma}_n^{-1}$ transforms (8.1) into (8.10). In general, the measurement matrices may not all be of rank M . If they are not, then even optimal Kalman filters, derived from the state-space model (8.2), will be effective only if the pairs $(\mathbf{H}_n, \mathbf{A}_n)$ satisfy the technical condition of *stochastic observability* [148]. This condition is usually satisfied because of the presence of delay or difference terms in \mathbf{x}_n . Any such (application dependent) *a priori* information should be incorporated into the nonparametric paradigm presented here.

The Solution

Again, by the Euler equations for (8.11), the solution $\hat{\mathbf{g}}_{\alpha}$ is a vector spline with component functions $\{\hat{g}_{m,\alpha}\}_{m=1}^M$ that are each natural cubic-splines. Their piecewise polynomial expansions (8.4) have coefficients denoted by $\hat{\mathbf{a}}^{(m)}$, $\hat{\mathbf{b}}^{(m)}$, $\hat{\mathbf{c}}^{(m)}$, and $\hat{\mathbf{d}}^{(m)}$, where

$$\begin{aligned}\hat{\mathbf{a}}^{(m)} &= (\hat{a}_1^{(m)}, \dots, \hat{a}_N^{(m)})', \\ \hat{\mathbf{c}}^{(m)} &= (\hat{c}_2^{(m)}, \dots, \hat{c}_{N-1}^{(m)})', \quad \hat{c}_1^{(m)} = \hat{c}_N^{(m)} = 0,\end{aligned}$$

and the coefficients $\hat{\mathbf{b}}^{(m)}$ and $\hat{\mathbf{d}}^{(m)}$ satisfy (8.6).

As shown in Appendix A, $\hat{\mathbf{a}}$ and $\hat{\mathbf{c}}$ are computed in $O(M^3N)$ operations by solving the following banded equations (cf. (8.9)):

$$\begin{aligned}(\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{y} &= ((\mathbf{T} \otimes \mathbf{D}(\alpha)^{-1}) + (\mathbf{Q}' \otimes \mathbf{I}_M)\boldsymbol{\Sigma}(\mathbf{Q} \otimes \mathbf{I}_M))\hat{\mathbf{c}}\alpha, \\ \hat{\mathbf{a}} &= \mathbf{y} - \boldsymbol{\Sigma}(\mathbf{Q} \otimes \mathbf{I}_M)\hat{\mathbf{c}}\alpha,\end{aligned}\tag{8.12}$$

where \otimes denotes matrix tensor product, \mathbf{I}_M is an $M \times M$ identity matrix,

$$\begin{aligned}\mathbf{a} &= (a_1^{(1)}, \dots, a_1^{(M)}, \dots, a_N^{(1)}, \dots, a_N^{(M)})', \\ \mathbf{c} &= (c_2^{(1)}, \dots, c_2^{(M)}, \dots, c_{N-1}^{(1)}, \dots, c_{N-1}^{(M)})', \\ \mathbf{c}\alpha &= (\mathbf{I}_{M(N-2)} \otimes \mathbf{D}(\alpha))\mathbf{c}, \\ \mathbf{D}(\alpha) &= \text{diag}(\alpha_1, \dots, \alpha_M), \\ \mathbf{y} &= (\mathbf{y}'_1, \dots, \mathbf{y}'_N)'\end{aligned}\tag{8.13}$$

and

$$\boldsymbol{\Sigma} = \text{diag}(\boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_N).$$

The matrix $\boldsymbol{\Sigma}$ is the $NM \times NM$ block diagonal covariance of \mathbf{y} .

The minimization (8.11) has resulted in a linear relationship, $\hat{\mathbf{a}} = \mathbf{A}(\alpha)\mathbf{y}$, between the measurements and the estimates, where from (8.12),

$$\mathbf{A}(\alpha) = \mathbf{I}_{NM} - \boldsymbol{\Sigma}(\mathbf{Q} \otimes \mathbf{I}_M)((\mathbf{T} \otimes \mathbf{D}(\alpha)^{-1}) + (\mathbf{Q}' \otimes \mathbf{I}_M)\boldsymbol{\Sigma}(\mathbf{Q} \otimes \mathbf{I}_M))^{-1}(\mathbf{Q}' \otimes \mathbf{I}_M).\tag{8.14}$$

In statistics, $\mathbf{A}(\alpha)$ is called the *hat* or *influence* matrix and will be used in the next section for automatic selection of α .

8.4 Choosing the Smoothing Parameters

If the smoothing parameter α is too large or too small, then the measurements will be over-smoothed or under-smoothed, respectively. In the scalar case, Reinsch [126] suggested choosing α so that $\text{RSS}(\hat{g}_{\alpha}) \approx N\sigma^2$.

However, Craven and Wahba [149] showed that this led to consistent over-smoothing. Ideally, we would like to choose the smoothing parameters to minimize the mean square error:

$$\begin{aligned} \text{MSE}(\boldsymbol{\alpha}) &\triangleq \frac{1}{N} \sum_{n=1}^N \|\hat{\mathbf{g}}_{\boldsymbol{\alpha}}(t_n) - \mathbf{g}(t_n)\|^2, \\ \boldsymbol{\alpha}_{\text{MSE}} &\triangleq \arg \min_{\boldsymbol{\alpha}} \text{MSE}(\boldsymbol{\alpha}). \end{aligned} \quad (8.15)$$

In practice, this minimization is impossible since \mathbf{g} is unknown. For the scalar measurement problem, several methods have been suggested for estimating $\boldsymbol{\alpha}_{\text{MSE}}$ from the data [127], including two due to Akaike [150]. We present below three of these methods, each generalized to apply to our vector measurement problem. They are compared empirically in Section 8.7. Note that for small samples one may prefer to use robust variants of these estimators [150].

The estimators discussed below all depend on the central bands of the influence matrix $\mathbf{A}(\boldsymbol{\alpha})$. Hutchinson and de Hoog [151, 152] presented algorithms for computing these bands in $O(M^3N)$ operations. Their algorithm is directly applicable to the vector measurement problem, so we do not present it here.

8.4.1 Unbiased Risk

In the scalar case with known error standard deviation, Craven and Wahba [149] have suggested using the value of the smoothing parameter that minimizes an unbiased estimator of the expected mean square error (risk). This idea can be extended directly to the vector measurement case. One can easily show that

$$\text{UR}(\boldsymbol{\alpha}) \triangleq \frac{1}{N} \|(\mathbf{I} - \mathbf{A}(\boldsymbol{\alpha}))\mathbf{y}\|^2 - \frac{2}{N} \text{trace}(\boldsymbol{\Sigma}(\mathbf{I} - \mathbf{A}(\boldsymbol{\alpha}))) + \frac{1}{N} \text{trace}(\boldsymbol{\Sigma})$$

is an unbiased estimator of $E\{\text{MSE}(\boldsymbol{\alpha})\}$. The unbiased risk estimate of $\boldsymbol{\alpha}_{\text{MSE}}$ is thus

$$\boldsymbol{\alpha}_{\text{UR}} \triangleq \arg \min_{\boldsymbol{\alpha}} \text{UR}(\boldsymbol{\alpha}).$$

The estimators $\boldsymbol{\alpha}_{\text{CV}}$ and $\boldsymbol{\alpha}_{\text{GCV}}$ discussed below have been more popular than $\boldsymbol{\alpha}_{\text{UR}}$ in the scalar case, perhaps because the latter depends on the (often unknown) error variance σ^2 . For the vector measurement problem, all three estimates depend on the covariance matrices $\{\boldsymbol{\Sigma}_n\}$.

8.4.2 Cross Validation

Wahba and Wold [110, 153] have suggested using the smoothing parameter that minimizes the cross-validation (CV) score:

$$\begin{aligned} \text{CV}(\boldsymbol{\alpha}) &\triangleq \frac{1}{N} \sum_{n=1}^N (\mathbf{y}_n - \hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}(t_n))' \boldsymbol{\Sigma}_n^{-1} (\mathbf{y}_n - \hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}(t_n)), \\ \boldsymbol{\alpha}_{\text{CV}} &\triangleq \arg \min_{\boldsymbol{\alpha}} \text{CV}(\boldsymbol{\alpha}). \end{aligned} \quad (8.16)$$

$\hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}$ is the solution to the smoothing problem (8.11) with $N-1$ data points, posed without the data pair (t_n, \mathbf{y}_n) , i.e.:

$$\hat{\mathbf{g}}_{\boldsymbol{\alpha},-i} \triangleq \arg \min_{\mathbf{g}} \left[\sum_{n=1, n \neq i}^N (\mathbf{y}_n - \mathbf{g}(t_n))' \boldsymbol{\Sigma}_n^{-1} (\mathbf{y}_n - \mathbf{g}(t_n)) + \sum_{m=1}^M \alpha_m \int (\ddot{g}_m(t))^2 dt \right].$$

Each data pair is dropped in turn, the smoothed curve $\hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}$ is estimated, and the predicted value $\hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}(t_n)$ is compared with the unused measurement. If the CV score is small, then we have chosen the smoothing parameter that makes the estimated curve a good self predictor.

Although equation (8.16) illustrates the idea behind cross-validation, it is computationally inefficient. We show in Appendix B that (8.16) can be rewritten

$$\text{CV}(\boldsymbol{\alpha}) = \frac{1}{N} \sum_{n=1}^N \|\boldsymbol{\Sigma}_n^{-\frac{1}{2}} (\mathbf{I}_M - \mathbf{A}_{(nn)}(\boldsymbol{\alpha}))^{-1} (\mathbf{y}_n - \hat{\mathbf{g}}_{\boldsymbol{\alpha}}(t_n))\|^2, \quad (8.17)$$

where $\mathbf{A}_{(nn)}(\boldsymbol{\alpha})$ is the n^{th} $M \times M$ block diagonal submatrix of the influence matrix (8.14). By using the Hutchinson and de Hoog algorithm [151], (8.17) is computed in only $O(M^3 N)$ operations.

8.4.3 Generalized Cross-Validation

Craven and Wahba [149] have also suggested using the smoothing parameter that minimizes the generalized cross-validation (GCV) score, whose vector spline generalization is:

$$\begin{aligned} \text{GCV}(\boldsymbol{\alpha}) &\triangleq \frac{\frac{1}{N} \text{RSS}(\hat{\mathbf{g}}_{\boldsymbol{\alpha}})}{(\frac{1}{N} \text{trace}(\mathbf{I} - \mathbf{A}(\boldsymbol{\alpha})))^2} = \frac{\frac{1}{N} (\mathbf{y} - \hat{\mathbf{g}}_{\boldsymbol{\alpha}})' \boldsymbol{\Sigma}^{-1} (\mathbf{y} - \hat{\mathbf{g}}_{\boldsymbol{\alpha}})}{(\frac{1}{N} \text{trace}(\mathbf{I} - \mathbf{A}(\boldsymbol{\alpha})))^2}, \\ \boldsymbol{\alpha}_{\text{GCV}} &\triangleq \arg \min_{\boldsymbol{\alpha}} \text{GCV}(\boldsymbol{\alpha}). \end{aligned}$$

In the scalar case, the GCV score is a weighted version of the CV score that is invariant to rotations of the data when periodic end conditions are imposed [150]. See [127] for a discussion of the asymptotic properties of GCV. Again, [151] is used to evaluate $\text{GCV}(\boldsymbol{\alpha})$ in $O(M^3 N)$ operations.

8.5 Error Covariance Estimation

For the scalar measurement case with unknown error variance, Wahba [154] (see also discussion in [155]) proposed the following estimator:

$$\hat{\sigma}^2 = \frac{\|(\mathbf{I} - \mathbf{A}(\boldsymbol{\alpha}))\mathbf{y}\|^2}{\text{trace}(\mathbf{I} - \mathbf{A}(\boldsymbol{\alpha}))}, \quad (8.18)$$

where the denominator is the natural extension of “degrees of freedom” to nonparametric regression. For the vector measurement case with unknown error covariance, if the measurement errors are identically distributed, that is $\boldsymbol{\Sigma}_n = \underline{\boldsymbol{\Sigma}} \forall n$, then we can generalize this idea to estimate $\underline{\boldsymbol{\Sigma}}$ by the following algorithm:

Figure 8.1: True curves (-) and noisy measurements (o,*).

Figure 8.2: Comparison of MSE, UR, CV, and GCV for scalar measurements.

1. For each $m = 1, \dots, M$, smooth the m^{th} measurement component $\{y_{n,m}\}_{n=1}^N$ to compute \hat{g}_{m,α_m} using the scalar algorithm of Section 8.2, and using, for example, the CV score to choose α_m .
2. Estimate the elements of $\underline{\Sigma}$ using the standard correlation estimate:

$$\hat{\underline{\Sigma}}_{ij} = \frac{\sum_{n=1}^N (y_{n,i} - \hat{g}_{i,\alpha_i}(t_n))(y_{n,j} - \hat{g}_{j,\alpha_j}(t_n))}{\sqrt{\text{trace}(\mathbf{I} - \mathbf{A}(\alpha_i))} \sqrt{\text{trace}(\mathbf{I} - \mathbf{A}(\alpha_j))}},$$

(which simplifies to (8.18) for the diagonal elements of $\underline{\Sigma}$). For the non-i.i.d. case, Silverman's iterative reweighting approach may be useful [108].

8.6 Algorithm

Table 8.2 outlines the organization of the algorithm's implementation (C source code is available as `vspline` from `netlib` [143]). The first set of computations are independent of α . The second set computes the smoothed estimates $\hat{\mathbf{a}}$, and the third set evaluates the cross-validation score. The computational requirements for this algorithm are of the same order as those for Kalman filter smoothers [156], when α is known. The second and third set are typically repeated for several values of α to find the minimum CV score. This search is the computational penalty for our uncertainty about the smoothness of the functions we are estimating. The computational requirements are based on the operation counts given in Table 8.3, most of which follow from the algorithms given in [147]. For the i.i.d. case, the transformation method of Miller and Wegman [134] is preferable as it requires only $O(M^2N)$ computations for the transformations, and $O(MN)$ computations for smoothing.

8.7 Simulation Results

To demonstrate the new smoothing algorithm and to compare the three methods of choosing the smoothing parameter α automatically, we applied the methods to simulated data. Figure 8.1 is a plot of two functions, g_1 , a decaying sinusoid, and g_2 , a hyperbolic tangent, and one realization of their noisy sampled measurements. We generated the measurements by adding pseudo-random Gaussian noise vectors with covariance matrix

$$\underline{\Sigma} = \begin{bmatrix} 2.25 & 2.4 \\ 2.4 & 4 \end{bmatrix}$$

Computation	Flops
I. Initialization	
\mathbf{Q}, \mathbf{T}	$N(7) - 7$
$(\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{y}$	$N(5M) - 5M$
$\mathbf{B}_0 = (\mathbf{Q}' \otimes \mathbf{I}_M)\boldsymbol{\Sigma}(\mathbf{Q} \otimes \mathbf{I}_M)$	$N(9M^2 + 6) - 23M^2 - 16$
Subtotal: $N(9M^2 + 5M + 13) - 23M^2 - 5M - 23$	
II. Regression	
$(\mathbf{T} \otimes \mathbf{D}(\boldsymbol{\alpha})^{-1})$	$N(2M) - 5M$
$\mathbf{B} = (\mathbf{T} \otimes \mathbf{D}(\boldsymbol{\alpha})^{-1}) + \mathbf{B}_0$	$N(2M^2) - 4M^2$
$[\mathbf{L}, \mathbf{D}] = \text{Cholesky}(\mathbf{B})$	$N(\frac{27}{2}M^3 - \frac{3}{2}M^2 - M)$ $-54M^3 + 12M^2 + 2M$
Solve $(\mathbf{LDL}')\hat{\mathbf{c}}\boldsymbol{\alpha} = (\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{y}$	$N(12M^2 - 3M) - 42M^2 + 12M$
$(\mathbf{Q}' \otimes \mathbf{I}_M)\hat{\mathbf{c}}\boldsymbol{\alpha}$	$N(6M) - 12M$
$\boldsymbol{\Sigma}(\mathbf{Q}' \otimes \mathbf{I}_M)\hat{\mathbf{c}}\boldsymbol{\alpha}$	$N(2M^2 - M)$
$\hat{\mathbf{a}} = \mathbf{y} - \boldsymbol{\Sigma}(\mathbf{Q}' \otimes \mathbf{I}_M)\hat{\mathbf{c}}\boldsymbol{\alpha}$	$N(M)$
Subtotal: $N(\frac{27}{2}M^3 + \frac{29}{2}M^2 + 4M) - 54M^3 - 34M^2 - 3M$	
III. Compute CV Score	
$\mathbf{B}^{-1} = \text{Invert}(\mathbf{LDL}') [151]$	$N(18M^3 - 6M^2) - 2M(3M - 1)(12M - 1)$
$\mathbf{F} = (\mathbf{Q} \otimes \mathbf{I}_M)\mathbf{B}^{-1}(\mathbf{Q}' \otimes \mathbf{I}_M)$	$N(\frac{27}{2}M(M + 1)) - \frac{39}{2}M(M + 1)$
$\mathbf{e}_n = \mathbf{y}_n - \hat{\mathbf{g}}\boldsymbol{\alpha}(t_n)$	$N(M)$
Solve $(\mathbf{F}_{(nn)})\mathbf{f}_n = \boldsymbol{\Sigma}_n^{-1}\mathbf{e}_n$	$N(\frac{1}{2}M^3 + \frac{5}{2}M^2 - M)$
$\text{CV}(\boldsymbol{\alpha}) = \frac{1}{N} \sum_{n=1}^N \mathbf{f}_n' \boldsymbol{\Sigma}_n^{-1} \mathbf{f}_n$	$N(2M^2 + M - 1)$
Subtotal: $N(\frac{37}{2}M^3 + 11M^2 + \frac{29}{2}M - 1) - 72M^3 + \frac{21}{2}M^2 - \frac{43}{2}M$	

Table 8.2: Computational requirements of linear smoothing algorithm.

Computation	Flops
Kron (J, n) with $\text{diag}(M)$	$n(M(J + 1)) - \frac{M}{2}J(J + 1)$
Kron (J, n) with $M \times M$	$n(M^2J + \frac{1}{2}M(M - 1)) - \frac{1}{2}M^2(J^2 + J)$
Chol (J, n)	$n(\frac{3}{2}J^2 + \frac{5}{2}J) - J(J + 1)^2$
Solve (J, n)	$n(1 + 4J) - 2J^2 - J$
Invert (J, n)	$n(2J(J + 1)) - \frac{2}{3}J(J + 1)(2J + 1)$

Table 8.3: Computational requirements for band matrix operations.

Figure 8.3: True curves (solid) and estimated curves (dashed) using CV score.

to $N = 100$ samples of the function drawn uniformly on $[0, 1]$.

First we applied the scalar smoothing algorithm to the noisy samples of g_1 shown in Figure 8.1. Figure 8.2 shows $\text{UR}(\alpha)$, $\text{CV}(\alpha)$, $\text{GCV}(\alpha)$, and $\text{MSE}(\alpha)$ over a range of the smoothing parameter. The minima of the UR, CV, and GCV scores (denoted by the small circles) occur very close to the minimum of the MSE; thus, at least for this scalar example, each of the three methods would select a good smoothing parameter. It is interesting that the UR, CV, and GCV scores are all flatter than the MSE near α_{MSE} .

To evaluate the three scores in the vector measurement case, we did a Monte Carlo simulation with 400 runs, each with a different measurement noise realization. For each run we computed α_{MSE} , α_{UR} , α_{CV} , and α_{GCV} using Powell's method for nonlinear minimization as given in [106, p. 315]. Our intent was to compare the estimators' best possible performances, so we initialized the minimization procedure at a value of α that resulted in low MSE for a few initial runs. To compare the estimators, we use their relative efficiencies, defined by:

$$\eta_i(\alpha) \triangleq \frac{\text{MSE}(\alpha_{\text{MSE},i})}{\text{MSE}(\alpha)},$$

where i indicates the i^{th} run. By definition (8.15), $\eta_i \in [0, 1]$.

Table 8.4 shows summary statistics of the computed relative efficiencies for the 400 runs. Three other cases are included for comparison; "None": no smoothing, " $\text{diag}(\underline{\Sigma})$ ": smoothing with just the diagonal components of the covariances (with minimum CV score), and " $\widehat{\underline{\Sigma}}$ ": smoothing with the estimated covariance procedure described in Section 8.5 (also with minimum CV score).

From the summary statistics for this example, we conclude that the CV and GCV scores perform equally well, and both slightly outperform the UR score. Those three were significantly more efficient than smoothing the components individually, which was expected since the measurement correlation was 0.8. All smoothing approaches decreased the MSE by a factor of approximately 10. It was a pleasant surprise that the performance using the estimated covariance matrix was about as good as the performance using the true covariance. This suggests that the approach described in this chapter may be preferable to smoothing the components individually, even when the error covariance is unknown. The off-diagonal elements of the covariance matrices clearly play an important role, even when estimated.

There is still no consensus on the relative theoretical merits of the UR, CV, and GCV scores, even in the scalar case. We have derived and presented the vector generalizations of all three since their performances may be application dependent.

As a representative example, Figure 8.3 shows the smoothed estimates (dashed) superimposed on the true curves (solid) for the data shown in Figure 8.1, using α_{CV} . The estimated functions agree well with the true functions, and the overall smoothness is qualitatively similar as well.

Summary statistic of $\{\eta_i\}_{i=1}^{400}$	Smoothing approach					
	None	UR	CV	GCV	$\text{diag}(\underline{\Sigma})$	$\hat{\underline{\Sigma}}$
mean $\bar{\eta}$	0.076	0.812	0.841	0.846	0.754	0.839
5 th %-ile	0.034	0.532	0.578	0.605	0.463	0.555
25 th %-ile	0.054	0.731	0.762	0.767	0.664	0.757
50 th %-ile	0.071	0.836	0.877	0.877	0.777	0.883
75 th %-ile	0.092	0.924	0.947	0.945	0.868	0.946
95 th %-ile	0.130	0.991	0.992	0.990	0.945	0.993
$\sigma_{\bar{\eta}} = \sigma_{\eta}/\sqrt{400}$	0.002	0.007	0.007	0.006	0.007	0.007

Table 8.4: Relative efficiencies of the different smoothing approaches.

8.8 Summary

We have presented a computationally efficient algorithm for nonparametric fixed-interval smoothing of noisy measurements with arbitrary measurement covariances. The effectiveness of the approach was demonstrated on a numerical example. The approach promises to be an attractive alternative to parametric Kalman smoothing for off-line applications.

Possible extensions of this work would include developing a more robust approach to covariance estimation, and applying Silverman's iterative reweighting approach [108] for non-i.i.d. covariance estimation. The relationship of nonparametric estimation to state-space methods could also be explored more completely, which could result in a recursive formulation of the solution.

8.9 Appendix A: Spline Smoothing Derivation

In this appendix, we derive the solution (8.12) to the minimization problem (8.11):

$$\hat{\mathbf{g}}\boldsymbol{\alpha} = \arg \min_{\mathbf{g}} \text{RSS}(\mathbf{g}) + \sum_{m=1}^M \alpha_m R(g_m).$$

By (8.13),

$$\text{RSS}(\mathbf{g}) = (\mathbf{y} - \mathbf{a})' \boldsymbol{\Sigma}^{-1} (\mathbf{y} - \mathbf{a}),$$

and by (8.8),

$$\sum_{m=1}^M \alpha_m R(g_m) = (\mathbf{c}^{(m)})' \mathbf{T} \mathbf{c}^{(m)} = \mathbf{c}' (\mathbf{T} \otimes \mathbf{D}(\boldsymbol{\alpha})) \mathbf{c}.$$

Since the minimizing component functions are each natural cubic-splines, they must each satisfy the constraint $\mathbf{Q}' \mathbf{a}^{(m)} = \mathbf{T} \mathbf{c}^{(m)}$ of (8.7). These constraints can be aggregated to form the constraint $(\mathbf{Q}' \otimes \mathbf{I}_M) \mathbf{a} = (\mathbf{T} \otimes \mathbf{I}_M) \mathbf{c}$. The optimal coefficients thus minimize

$$(\mathbf{y} - \mathbf{a})' \boldsymbol{\Sigma}^{-1} (\mathbf{y} - \mathbf{a}) + \mathbf{c}' (\mathbf{T} \otimes \mathbf{D}(\boldsymbol{\alpha})) \mathbf{c} \quad (8.19)$$

subject to

$$(\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{a} = (\mathbf{T} \otimes \mathbf{I}_M)\mathbf{c}. \quad (8.20)$$

Since $(\mathbf{T} \otimes \mathbf{I}_M)$ is symmetric and invertible, $\mathbf{c} = (\mathbf{T} \otimes \mathbf{I}_M)^{-1}(\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{a}$, which, substituted into (8.19) yields

$$\begin{aligned} & (\mathbf{y} - \mathbf{a})'\boldsymbol{\Sigma}^{-1}(\mathbf{y} - \mathbf{a}) + ((\mathbf{T} \otimes \mathbf{I}_M)^{-1}(\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{a})'(\mathbf{T} \otimes \mathbf{D}(\boldsymbol{\alpha}))(\mathbf{T} \otimes \mathbf{I}_M)^{-1}(\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{a} \\ & = \mathbf{y}'\boldsymbol{\Sigma}^{-1}\mathbf{y} - 2\mathbf{a}'\boldsymbol{\Sigma}^{-1}\mathbf{y} + \mathbf{a}'(\boldsymbol{\Sigma}^{-1} + (\mathbf{Q} \otimes \mathbf{D}(\boldsymbol{\alpha}))(\mathbf{T} \otimes \mathbf{I}_M)^{-1}(\mathbf{Q}' \otimes \mathbf{I}_M))\mathbf{a}. \end{aligned}$$

Minimizing over \mathbf{a} by setting the partial derivative with respect to \mathbf{a} equal to zero yields

$$\boldsymbol{\Sigma}^{-1}\mathbf{y} = (\boldsymbol{\Sigma}^{-1} + (\mathbf{Q} \otimes \mathbf{D}(\boldsymbol{\alpha}))(\mathbf{T} \otimes \mathbf{I}_M)^{-1}(\mathbf{Q}' \otimes \mathbf{I}_M))\hat{\mathbf{a}}. \quad (8.21)$$

Solving for $\hat{\mathbf{a}}$:

$$\hat{\mathbf{a}} = (\boldsymbol{\Sigma}^{-1} + \mathbf{S}\boldsymbol{\alpha})^{-1}\boldsymbol{\Sigma}^{-1}\mathbf{y}, \quad (8.22)$$

where

$$\mathbf{S}\boldsymbol{\alpha} \triangleq (\mathbf{Q} \otimes \mathbf{I}_M)(\mathbf{T} \otimes \mathbf{D}(\boldsymbol{\alpha})^{-1})^{-1}(\mathbf{Q}' \otimes \mathbf{I}_M) = (\mathbf{Q}\mathbf{T}^{-1}\mathbf{Q}') \otimes \mathbf{D}(\boldsymbol{\alpha}). \quad (8.23)$$

We could compute $\hat{\mathbf{a}}$ directly from equation (8.22), but a few manipulations [126] yield a banded form that is easier to evaluate. Multiplying both sides of (8.21) by $\boldsymbol{\Sigma}$ and using (8.20), we get

$$\mathbf{y} = \hat{\mathbf{a}} + \boldsymbol{\Sigma}(\mathbf{Q} \otimes \mathbf{I}_M)(\mathbf{I}_{M(N-2)} \otimes \mathbf{D}(\boldsymbol{\alpha}))\hat{\mathbf{c}}.$$

Multiplying both sides by $(\mathbf{Q}' \otimes \mathbf{I}_M)\boldsymbol{\Sigma}$ and using (8.20) yields:

$$(\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{y} = ((\mathbf{T} \otimes \mathbf{I}_M) + (\mathbf{Q}' \otimes \mathbf{I}_M)\boldsymbol{\Sigma}(\mathbf{Q} \otimes \mathbf{I}_M)(\mathbf{I}_{M(N-2)} \otimes \mathbf{D}(\boldsymbol{\alpha})))\hat{\mathbf{c}}. \quad (8.24)$$

Symmetric band matrices are the easiest to use, so define $\mathbf{c}\boldsymbol{\alpha} = (\mathbf{I}_{M(N-2)} \otimes \mathbf{D}(\boldsymbol{\alpha}))\mathbf{c}$. Combining this definition with (8.24) yields:

$$(\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{y} = ((\mathbf{T} \otimes \mathbf{D}(\boldsymbol{\alpha})^{-1}) + (\mathbf{Q}' \otimes \mathbf{I}_M)\boldsymbol{\Sigma}(\mathbf{Q} \otimes \mathbf{I}_M))\hat{\mathbf{c}}\boldsymbol{\alpha}.$$

Thus the minimizing coefficients $\hat{\mathbf{a}}$ and $\hat{\mathbf{c}}$ are the solutions to the following system of equations:

$$((\mathbf{T} \otimes \mathbf{D}(\boldsymbol{\alpha})^{-1}) + (\mathbf{Q}' \otimes \mathbf{I}_M)\boldsymbol{\Sigma}(\mathbf{Q} \otimes \mathbf{I}_M))\hat{\mathbf{c}}\boldsymbol{\alpha} = (\mathbf{Q}' \otimes \mathbf{I}_M)\mathbf{y},$$

$$\hat{\mathbf{c}} = (\mathbf{I}_{M(N-2)} \otimes \mathbf{D}(\boldsymbol{\alpha})^{-1})\hat{\mathbf{c}}\boldsymbol{\alpha},$$

$$\hat{\mathbf{a}} = \mathbf{y} - \boldsymbol{\Sigma}(\mathbf{Q} \otimes \mathbf{I}_M)\hat{\mathbf{c}}\boldsymbol{\alpha}.$$

8.10 Appendix B: CV Score Derivation

In this appendix, we show that equation (8.16) is equivalent to equation (8.17). Again we use $\mathbf{A}_{(nn)}(\boldsymbol{\alpha})$ to denote the n^{th} $M \times M$ central diagonal submatrix of $\mathbf{A}(\boldsymbol{\alpha})$. The same arguments used in the proof of Lemma 3.1 by Craven and Wahba [149] imply that for all n and i

$$\hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}(t_i) = \sum_{k=1, k \neq n}^N \mathbf{A}_{(ik)}(\boldsymbol{\alpha}) \mathbf{y}_k + \mathbf{A}_{(in)}(\boldsymbol{\alpha}) \hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}(t_n).$$

By the definition of $\mathbf{A}(\boldsymbol{\alpha})$,

$$\begin{aligned} \mathbf{y}_n - \hat{\mathbf{g}}_{\boldsymbol{\alpha}}(t_n) &= \mathbf{y}_n - \sum_{i=1}^N \mathbf{A}_{(ni)}(\boldsymbol{\alpha}) \mathbf{y}_i \\ &= \mathbf{y}_n - \sum_{i=1, i \neq n}^N \mathbf{A}_{(ni)}(\boldsymbol{\alpha}) \mathbf{y}_i - \mathbf{A}_{(nn)}(\boldsymbol{\alpha}) \hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}(t_n) - \mathbf{A}_{(nn)}(\boldsymbol{\alpha}) (\mathbf{y}_n - \hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}(t_n)) \\ &= \mathbf{y}_n - \hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}(t_n) - \mathbf{A}_{(nn)}(\boldsymbol{\alpha}) (\mathbf{y}_n - \hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}(t_n)) \\ &= (\mathbf{I} - \mathbf{A}_{(nn)}(\boldsymbol{\alpha})) (\mathbf{y}_n - \hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}(t_n)). \end{aligned}$$

Therefore

$$\mathbf{y}_n - \hat{\mathbf{g}}_{\boldsymbol{\alpha},-n}(t_n) = (\mathbf{I} - \mathbf{A}_{(nn)}(\boldsymbol{\alpha}))^{-1} (\mathbf{y}_n - \hat{\mathbf{g}}_{\boldsymbol{\alpha}}(t_n)),$$

which can be substituted into (8.16) to yield (8.17).

Chapter 9

Nonlinear Nonparametric Smoothing

This chapter generalizes the linear smoothing algorithm of Chapter 8 to the problem of estimating a smooth vector-valued function given noisy nonlinear vector-valued measurements of that function¹. We present a nonparametric optimality criterion for this estimation problem, and develop a computationally efficient iterative algorithm for its solution. The new algorithm provides an alternative to the extended Kalman filter, as it does not require a parametric state-space model. We also present an automatic procedure that uses the measurements to determine how much to smooth. The preceding chapters have demonstrated the performance of this algorithm on the object-estimation problem; here, the algorithm demonstrates subpixel estimation accuracy on a problem from picture processing: estimation of a curved edge in a noisy image.

9.1 Introduction

This chapter considers the problem of estimating a smooth vector-valued function from noisy measurements observed through a nonlinear mapping. We assume the following nonlinear measurement model:

$$\mathbf{y}_n = \mathbf{h}_n(\mathbf{x}_n) + \varepsilon_n, \quad n = 1, \dots, N, \quad (9.1)$$

where

$$\varepsilon_n, \mathbf{y}_n \in \mathfrak{R}^{L_n}, \quad \mathbf{x}_n \in \mathfrak{R}^M, \quad \text{and } \mathbf{h}_n : \mathfrak{R}^M \rightarrow \mathfrak{R}^{L_n}.$$

We assume the additive measurement errors are independent between samples and are normally distributed with mean zero. Without loss of generality, we assume the covariance matrix of ε_n is $\sigma^2 \mathbf{I}$, where σ^2 may be

¹This chapter is derived largely from [129].

unknown². The states $\{\mathbf{x}_n\}$ are (possibly unequally spaced) samples of a smooth vector-valued function \mathbf{g} :

$$\mathbf{x}_n = [g_1(t_n), \dots, g_M(t_n)]' \triangleq \mathbf{g}(t_n), \quad t_n < t_{n+1} \quad \forall n, \quad (9.2)$$

where “ $'$ ” denotes matrix transposition. The goal is to estimate \mathbf{g} from the measurements $\{\mathbf{y}_n\}_{n=1}^N$.

The prevalent approach to this estimation problem is the extended Kalman filter (EKF) [132]. The EKF hinges on an assumption that the states adhere to a parametric Gauss-Markov state-space model. However, in applications such as the edge-estimation example given in Section 9.6, the parameters required by the EKF formulation (state evolution matrices and covariances) are unknown and are difficult to determine. Furthermore, the state-space formulae imply the *a priori* variance of the function varies with t . Although it is natural for tracking applications, where one is given a starting state that evolves with increasing uncertainty over time, this variation is counter-intuitive for off-line applications such as image processing, where t often represents *space* rather than *time*. For example, when detecting and estimating an edge in an image, the *a priori* variance of the position of the edge (the uncertainty before actually seeing the image) is the same throughout the image. Despite these objections to parametric methods, we must use our *a priori* knowledge of the smoothness of the underlying functions if we are to obtain accurate estimates. This necessity has motivated nonparametric approaches to smoothing [108, 127], and is the basis for the new algorithm presented in this chapter.

In Chapter 8, we presented a computationally efficient algorithm for nonparametric smoothing for the special case when \mathbf{h}_n is linear, and we presented the rationale behind “penalized likelihood” estimation. Here, just as in the linear case, we must compromise between the agreement with the data and the smoothness of the estimated functions. Thus we propose the following optimality criterion:

$$\hat{\mathbf{g}} \triangleq \arg \min_{\mathbf{g}} \Phi_{\star}(\mathbf{g}),$$

$$\Phi_{\star}(\mathbf{g}) \triangleq \sum_{n=1}^N \|\mathbf{y}_n - \mathbf{h}_n(\mathbf{g}(t_n))\|^2 + \sum_{m=1}^M \alpha_m \int (g_m^{(k)}(t))^2 dt. \quad (9.3)$$

This criterion is the natural generalization of (8.11). O’Sullivan [157] considered this criterion for the case of scalar measurements. For simplicity, we assume $k = 2$. The parameter $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_M)$ controls the influence of the penalty term, and in Section 9.4 we describe how to estimate $\boldsymbol{\alpha}$ from the measurements automatically. Until then, we assume $\boldsymbol{\alpha}$ is known.

By the “minimal property of splines” proven in Theorem 2 of [125], any function $\hat{\mathbf{g}}$ that achieves the minimum of Φ_{\star} is a vector spline with component functions that are cubic splines (for $k = 2$). (We restrict our attention here to natural cubic-splines by imposing the end conditions that $g_m(t)$ is linear for $t < t_1$ and $t > t_N$.) However, unlike in the linear case, in general there may be multiple minima³. Physical constraints

²If the measurement error has the (positive definite) covariance matrix $\sigma^2 \boldsymbol{\Sigma}_n$, then we can premultiply \mathbf{y}_n and \mathbf{h}_n by $\boldsymbol{\Sigma}_n^{-1/2}$. Singular covariances may be the result of linearly dependent measurements, indicating that other constraints should be incorporated.

³Consider $\mathbf{h}_n(x) = x^2$, then $\Phi_{\star}(g(t)) = \Phi_{\star}(-g(t))$.

will usually rule out the irrelevant solutions. The EKF suffers the same ambiguity, a fact usually ignored since the filter update equations are initialized at some (presumably meaningful) starting state. The iterative algorithms we present below also require an initial estimate.

Since the component functions of $\hat{\mathbf{g}}$ are natural cubic-splines, we need only estimate the coefficients of their piecewise-polynomial expansions (or, for better numerical stability, their B-spline expansion [145]). In fact, if we compute $\hat{\mathbf{g}}(t)$ at t_1, \dots, t_N , then we can compute all the coefficients from (8.6). From (9.2), this is equivalent to estimating the states $\{\mathbf{x}_n\}_{n=1}^N$. From Section 8.9,

$$\sum_{m=1}^M \alpha_m \int (g_m^{(2)}(t))^2 dt = \mathbf{x}' \mathbf{S}_\alpha \mathbf{x},$$

where \mathbf{S}_α is defined by (8.23), and

$$\mathbf{x} \triangleq [\mathbf{x}'_1, \dots, \mathbf{x}'_N]'$$

Therefore, the variational problem (9.3) is equivalent to the following penalized nonlinear least-squares problem:

$$\begin{aligned} \hat{\mathbf{x}}_\alpha &= \arg \min_{\mathbf{x}} \Phi_\alpha(\mathbf{x}), \\ \Phi_\alpha(\mathbf{x}) &= \|\mathbf{y} - \mathbf{h}(\mathbf{x})\|^2 + \mathbf{x}' \mathbf{S}_\alpha \mathbf{x}, \end{aligned}$$

where

$$\mathbf{y} \triangleq [\mathbf{y}'_1, \dots, \mathbf{y}'_N]', \quad \mathbf{h}(\mathbf{x}) \triangleq [\mathbf{h}_1(\mathbf{x}_1)', \dots, \mathbf{h}_N(\mathbf{x}_N)']'$$

\mathbf{S}_α , which also depends on k in general, is the spline penalty matrix that prohibits excessive local variation in $\hat{\mathbf{g}}$.

In Sections 9.2 and 9.3, we develop an iterative method for computing $\hat{\mathbf{x}}_\alpha$. This method is summarized as a computationally efficient algorithm in Section 9.5, after we discuss selection of α in Section 9.4. We demonstrate the algorithm on a curved-edge estimation problem in Section 9.6, and conclude with open problems in Section 9.7.

9.2 Linearization Approach

We first consider estimating \mathbf{x} by a linearization method similar to the EKF approach. Assume $\mathbf{x}_o \triangleq [\mathbf{x}'_{o,1}, \dots, \mathbf{x}'_{o,N}]'$ is an initial estimate⁴ of \mathbf{x} . By the first-order Taylor's expansion of \mathbf{h}_n about $\mathbf{x}_{o,n}$:

$$\mathbf{h}_n(\mathbf{x}_n) \approx \mathbf{h}_n(\mathbf{x}_{o,n}) + \mathbf{H}_n(\mathbf{x}_n - \mathbf{x}_{o,n}), \quad (9.4)$$

where \mathbf{H}_n is the $L_n \times M$ Jacobian of \mathbf{h}_n evaluated at $\mathbf{x}_{o,n}$. Substituting (9.4) into (9.1), we get

$$\mathbf{y}_n \approx \mathbf{h}_n(\mathbf{x}_{o,n}) + \mathbf{H}_n(\mathbf{x}_n - \mathbf{x}_{o,n}) + \boldsymbol{\varepsilon}_n.$$

Multiplying both sides by $(\mathbf{H}'_n \mathbf{H}_n)^{-1} \mathbf{H}'_n$ and rearranging yields

$$\mathbf{z}_n \approx \mathbf{x}_n + \mathbf{v}_n, \quad (9.5)$$

where

$$\mathbf{z}_n \triangleq \mathbf{x}_{o,n} + (\mathbf{H}'_n \mathbf{H}_n)^{-1} \mathbf{H}'_n (\mathbf{y}_n - \mathbf{h}_n(\mathbf{x}_{o,n})).$$

The transformed measurement error \mathbf{v}_n is normally distributed with mean zero and covariance matrix $\mathbf{\Pi}_n = (\mathbf{H}'_n \mathbf{H}_n)^{-1}$. This procedure requires⁵ that $(\mathbf{H}'_n \mathbf{H}_n)$ be invertible, or equivalently that the Jacobians all have rank M . A necessary condition is therefore that $L_n \geq M \forall n$. (One special case is worth noting: if the initial estimate is the (unpenalized) maximum-likelihood estimate, i.e., $\mathbf{x}_{o,n} = \arg \min_{\mathbf{x}_n} \|\mathbf{y}_n - \mathbf{h}_n(\mathbf{x}_n)\|^2$, then $\mathbf{z}_n = \mathbf{x}_{o,n}$, and each covariance matrix $\mathbf{\Pi}_n$ is a corresponding Fisher information matrix.)

We have transformed the nonlinear measurements (9.1) into a set of linearized measurements (9.5) that are now in a form suitable for the linear vector-spline smoothing algorithm of Chapter 8. The resulting estimate, denoted by $\hat{\mathbf{x}}_{\text{Lin}}$, satisfies

$$\begin{aligned} \hat{\mathbf{x}}_{\text{Lin}} &= \arg \min_{\mathbf{x}} \Phi_o(\mathbf{x}), \\ \Phi_o(\mathbf{x}) &= (\mathbf{z} - \mathbf{x})' \mathbf{\Pi}^{-1} (\mathbf{z} - \mathbf{x}) + \mathbf{x}' \mathbf{S}_\alpha \mathbf{x}, \end{aligned}$$

where

$$\mathbf{z} \triangleq [\mathbf{z}'_1, \dots, \mathbf{z}'_N]', \quad \mathbf{\Pi} = \text{diag}(\mathbf{\Pi}_n).$$

Φ_o is a quadratic form, and its minimizer (cf. (8.22)) is:

$$\hat{\mathbf{x}}_{\text{Lin}} = (\mathbf{\Pi}^{-1} + \mathbf{S}_\alpha)^{-1} \mathbf{\Pi}^{-1} \mathbf{z}. \quad (9.6)$$

In the implementation of this algorithm, we compute $\hat{\mathbf{x}}_{\text{Lin}}$ with the $O(M^3 N)$ algorithm of Chapter 8, rather than by direct evaluation of (9.6).

⁴Obtaining an initial estimate is clearly application dependent. The transform approach of Bresler [158] is well suited for nonlinearities that separate into "shift" and "shape" parameters.

⁵It is not strictly necessary that all the Jacobians exist or have rank M . Spline smoothing can be applied to non-uniformly spaced measurements, so one could simply discard any measurements violating the existence or rank conditions.

A significant difference between this nonparametric approach and the EKF approach is the timing of the linearization. Here, we first independently linearize all the measurements, and then smooth. For the EKF, the measurements are linearized about the most recent estimate from the recursive update formulae [132]. Though our approach therefore requires more effort “up front” in obtaining initial estimates, it does avoid some of the potential problems of EKF mistracking [78, 158].

Since the accuracy of $\hat{\mathbf{x}}_{\text{Lin}}$ depends on the accuracy of (9.4), we would usually iterate by using $\hat{\mathbf{x}}_{\text{Lin}}$ as a new “initial” estimate and repeating the above procedure. Unfortunately, there is no guarantee such iterations will accomplish our original goal of minimizing Φ_* or will even converge. The most we can claim is that the optimal estimate $\hat{\mathbf{x}}_{\alpha}$ is a fixed-point of the iterations, i.e., if $\mathbf{x}_o = \hat{\mathbf{x}}_{\alpha}$ then $\hat{\mathbf{x}}_{\text{Lin}} = \hat{\mathbf{x}}_{\alpha}$. The standard solution to this dilemma is to introduce a relaxation parameter. It is not clear how to do this from the above derivation, despite its intuitive appeal. With an eye towards applying the Levenberg-Marquardt relaxation method [159], in the next section we derive the Hessian estimate of \mathbf{x} .

9.3 Hessian Approach

The Hessian approach [106] for nonlinear least-squares problems is to approximate the functional Φ_{α} locally by a quadratic:

$$\Phi_{\alpha}(\mathbf{x}) \approx \Phi_{\alpha}(\mathbf{x}_o) - 2\mathbf{d}'(\mathbf{x} - \mathbf{x}_o) + (\mathbf{x} - \mathbf{x}_o)'\mathbf{D}(\mathbf{x} - \mathbf{x}_o).$$

The estimate is then given as

$$\hat{\mathbf{x}}_{\text{Hess}} = \mathbf{x}_o + \boldsymbol{\delta},$$

where $\boldsymbol{\delta}$ is the solution to

$$\mathbf{D}\boldsymbol{\delta} = \mathbf{d}. \tag{9.7}$$

Neglecting second derivatives [106], one can easily compute \mathbf{d} and \mathbf{D} :

$$\mathbf{d} = -\frac{1}{2} \left. \frac{\partial \Phi_{\alpha}}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_o} = \mathbf{H}'(\mathbf{y} - \mathbf{h}(\mathbf{x}_o)) - \mathbf{S}_{\alpha}\mathbf{x}_o,$$

and

$$\mathbf{D} = \frac{1}{2} \left\{ \frac{\partial \Phi_{\alpha}}{\partial \mathbf{x}_i} \frac{\partial \Phi_{\alpha}}{\partial \mathbf{x}_j} \right\} = (\boldsymbol{\Pi}^{-1} + \mathbf{S}_{\alpha}),$$

where

$$\mathbf{H} \triangleq \text{diag}(\mathbf{H}_n).$$

Therefore, the Hessian estimate is:

$$\hat{\mathbf{x}}_{\text{Hess}} = \mathbf{x}_o + (\boldsymbol{\Pi}^{-1} + \mathbf{S}_{\alpha})^{-1}[\mathbf{H}'(\mathbf{y} - \mathbf{h}(\mathbf{x}_o)) - \mathbf{S}_{\alpha}\mathbf{x}_o]. \tag{9.8}$$

The Levenberg-Marquardt (LM) approach [159] to relaxation of the Hessian nonlinear least-squares method is equally applicable to our penalized nonlinear least-squares problem, since the penalty is a quadratic. Instead of (9.7), the LM approach (see discussion in [106, §14.4]) is to compute the update as follows:

$$(\mathbf{D} + \lambda\mathbf{\Lambda})\boldsymbol{\delta}_\lambda = \mathbf{d},$$

yielding the estimate

$$\hat{\mathbf{x}}_{\text{Hess},\lambda} = \mathbf{x}_o + \boldsymbol{\delta}_\lambda.$$

The matrix $\mathbf{\Lambda}$ is diagonal, and its elements are a measure of scale; we take $\mathbf{\Lambda}$ to be the diagonal elements of $\mathbf{\Pi}^{-1}$. The LM algorithm provides a procedure for choosing the relaxation parameter λ to ensure that the new estimate is better than the previous estimate, i.e., $\Phi_{\boldsymbol{\alpha}}(\hat{\mathbf{x}}_{\text{Hess},\lambda}) < \Phi_{\boldsymbol{\alpha}}(\mathbf{x}_o)$. This procedure guarantees convergence to a local minimum when one iterates the Hessian method.

By applying (9.5), (9.6), and (9.8):

$$\begin{aligned} \hat{\mathbf{x}}_{\text{Lin}} &= (\mathbf{\Pi}^{-1} + \mathbf{S}\boldsymbol{\alpha})^{-1}\mathbf{\Pi}^{-1}\mathbf{z} \\ &= (\mathbf{\Pi}^{-1} + \mathbf{S}\boldsymbol{\alpha})^{-1}\mathbf{\Pi}^{-1}[\mathbf{\Pi}\mathbf{H}'(\mathbf{y} - \mathbf{h}(\mathbf{x}_o)) + \mathbf{x}_o] \\ &= (\mathbf{\Pi}^{-1} + \mathbf{S}\boldsymbol{\alpha})^{-1}[\mathbf{H}'(\mathbf{y} - \mathbf{h}(\mathbf{x}_o)) + \mathbf{\Pi}^{-1}\mathbf{x}_o + \mathbf{S}\boldsymbol{\alpha}\mathbf{x}_o - \mathbf{S}\boldsymbol{\alpha}\mathbf{x}_o] \\ &= \mathbf{x}_o + (\mathbf{\Pi}^{-1} + \mathbf{S}\boldsymbol{\alpha})^{-1}[\mathbf{H}'(\mathbf{y} - \mathbf{h}(\mathbf{x}_o)) - \mathbf{S}\boldsymbol{\alpha}\mathbf{x}_o] \\ &= \hat{\mathbf{x}}_{\text{Hess}}, \end{aligned}$$

we see that the Hessian approach and the linearization approach of Section 9.2 are equivalent, i.e., $\hat{\mathbf{x}}_{\text{Lin}} = \hat{\mathbf{x}}_{\text{Hess}}$. Using this equivalence, we can translate the relaxation parameter idea back into the spline-smoothing formulation. By the same arguments as above, if we define

$$\hat{\mathbf{x}}_{\text{Lin},\lambda} \triangleq (\mathbf{\Pi}_\lambda^{-1} + \mathbf{S}\boldsymbol{\alpha})^{-1}\mathbf{\Pi}_\lambda^{-1}\mathbf{z}_\lambda,$$

where

$$\mathbf{z}_\lambda \triangleq \mathbf{\Pi}_\lambda\mathbf{H}'(\mathbf{y} - \mathbf{h}(\mathbf{x}_o)) + \mathbf{x}_o,$$

and

$$\mathbf{\Pi}_\lambda^{-1} \triangleq \mathbf{\Pi}^{-1} + \lambda\mathbf{\Lambda},$$

then $\hat{\mathbf{x}}_{\text{Lin},\lambda} = \hat{\mathbf{x}}_{\text{Hess},\lambda}$. In words, rather than smoothing the pseudo-measurements \mathbf{z}_n (with covariances $\mathbf{\Pi}_n$), we smooth $\mathbf{z}_{n,\lambda}$ (with covariances $\mathbf{\Pi}_{n,\lambda}$). This estimation procedure is translated into an algorithm in Section 9.5.

9.4 Choosing the Smoothing Parameters

As in the linear case, we want to choose the smoothing parameter α to provide good estimates of \mathbf{g} . One method with intuitive appeal and high (statistical) efficiency (as shown in Chapter 8) in the linear case is to choose the smoothing parameter that minimizes the cross-validation (CV) score, defined by

$$\text{CV}(\alpha) \triangleq \frac{1}{N} \sum_{n=1}^N \|\mathbf{y}_n - \mathbf{h}_n(\hat{\mathbf{g}}_{\alpha, -n}(t_n))\|^2, \quad (9.9)$$

where

$$\hat{\mathbf{g}}_{\alpha, -i} \triangleq \arg \min_{\mathbf{g}} \sum_{n=1, n \neq i}^N \|\mathbf{y}_n - \mathbf{h}_n(\mathbf{g}(t_n))\|^2 + \sum_{m=1}^M \alpha_m \int (\ddot{g}_m(t))^2 dt.$$

$\hat{\mathbf{g}}_{\alpha, -i}$ is the solution to the smoothing problem posed without data point \mathbf{y}_i . Exact evaluation of (9.9) is impractical since it would require N iterative smoothing problems for each value of α . Motivated by the corresponding formula for the linear case (8.17), we propose the following approach: for a given value of α , compute $\hat{\mathbf{x}}_{\alpha}$, and use $\hat{\mathbf{x}}_{\alpha}$ to compute the linearized measurement \mathbf{z} , the Jacobian \mathbf{H} , and the covariance $\mathbf{\Pi} = (\mathbf{H}'\mathbf{H})^{-1}$. Then an approximation for $\text{CV}(\alpha)$ is

$$\text{CV}_0(\alpha) \triangleq \frac{1}{N} \sum_{n=1}^N \|\mathbf{\Pi}_n^{-\frac{1}{2}} (\mathbf{I}_M - \mathbf{A}_{(nn)}(\alpha))^{-1} (\mathbf{z}_n - \hat{\mathbf{x}}_{\alpha, n})\|^2, \quad (9.10)$$

where (cf. (9.6))

$$\mathbf{A}(\alpha) \triangleq (\mathbf{\Pi}^{-1} + \mathbf{S}\alpha)^{-1} \mathbf{\Pi}^{-1},$$

and $\mathbf{A}_{(nn)}(\alpha)$ is the n^{th} $M \times M$ block diagonal submatrix of $\mathbf{A}(\alpha)$. This approximation is based on the expectation that $\hat{\mathbf{x}}_{\alpha}$ will be close enough to \mathbf{g} that the Taylor expansion (9.4) will be accurate. Once $\hat{\mathbf{x}}_{\alpha}$ is computed, (9.10) is evaluated in $O(M^3N)$ operations as discussed in Chapter 8. The accuracy of the approximation used in deriving CV_0 is less important than whether or not the minimum of CV_0 occurs at a value of α for which $\hat{\mathbf{x}}_{\alpha}$ is a good estimate. In Section 9.6 we show an empirical result that indicates the utility of CV_0 .

9.5 Algorithm

The algorithm depicted in Table 9.1 computes $\hat{\mathbf{x}}_{\alpha}$ iteratively for a particular value of α . The computational complexity is only $O(M^3N)$. We have borrowed ideas from [106, §14.4], substituting in our optimality criteria. All operations containing terms with the subscript n are repeated for $n = 1, \dots, N$. Source code for this algorithm is available as `vspline` from `netlib` [143]. The dominant computational requirements are the vector-spline smoothing and the computation of CV_0 . Since these computations are required even in the linear measurement case, the principle “penalty” incurred when considering nonlinear problems is the necessity of iteration.

The algorithm of Table 9.1 is implemented as a procedure that returns $CV_0(\alpha)$. This procedure is typically called with several different values of α to minimize $CV_0(\alpha)$. We used the subroutine given in [106] for Powell's method for this minimization. We can make considerable computational savings by using the smoothed estimates for one value of α as the initial state when smoothing for a nearby value of α . Using this procedure, we have found empirically that although the smoothing algorithm may require six to ten iterations for the first value of α , on subsequent calls the smoothing procedure typically converges to within 0.1% of $\min_{\mathbf{x}} \Phi_{\alpha}(\mathbf{x})$ in just one or two steps.

In the examples of Section 9.6, the "else" section of this procedure rarely executed, hence the iterations converged nearly quadratically to the estimate $\hat{\mathbf{x}}_{\alpha}$.

Computation	Flops
Obtain initial estimate \mathbf{x}_0	?
$\hat{\mathbf{x}} := \mathbf{x}_0$	
$\lambda := 0.001$	
$\hat{\mathbf{y}}_n := \mathbf{h}_n(\hat{\mathbf{x}}_n)$?
$\mathbf{H}_n :=$ Jacobian of \mathbf{h}_n at $\hat{\mathbf{x}}_n$?
$\mathbf{e}_n := \mathbf{y}_n - \hat{\mathbf{y}}_n$	$N(L)$
$\mathbf{w}_n := \mathbf{H}'_n \mathbf{e}_n$	$N(2ML)$
$\mathbf{\Pi}_n^{-1} := (\mathbf{H}'_n \mathbf{H}_n)$	$N(2M^2L)$
$f_{\text{best}} := \Phi_{\boldsymbol{\alpha}}(\hat{\mathbf{x}})$	$N(14M)$
$\mathbf{\Lambda}_n :=$ diagonal elements of $\mathbf{\Pi}_n^{-1}$	
repeat {	
$\mathbf{\Pi}_{n,\lambda}^{-1} := \mathbf{\Pi}_n^{-1} + \lambda \mathbf{\Lambda}_n$	$N(2M)$
Invert $\mathbf{\Pi}_{n,\lambda}^{-1}$	$N(M^3)$
$\mathbf{z}_{n,\lambda} := \hat{\mathbf{x}}_n + \mathbf{\Pi}_{n,\lambda}^{-1} \mathbf{w}_n$	$N(2M^2)$
$\hat{\mathbf{x}}_{\text{Lin},\lambda} :=$ vector-spline smooth $\{\mathbf{z}_{n,\lambda}\}$, covariances $\{\mathbf{\Pi}_{n,\lambda}\}$	$N(\frac{27}{2}M^3)$
$\hat{\mathbf{y}}_n = \mathbf{h}_n(\hat{\mathbf{x}}_{\text{Lin},\lambda})$?
$\hat{\mathbf{e}}_n = \mathbf{y}_n - \hat{\mathbf{y}}_n$	$N(L)$
$f_{\text{new}} := \Phi_{\boldsymbol{\alpha}}(\hat{\mathbf{x}}_{\text{Lin},\lambda})$	$N(7M)$
if ($f_{\text{new}} < f_{\text{best}}$)	
$\hat{\mathbf{x}} := \hat{\mathbf{x}}_{\text{Lin},\lambda}$	
$f_{\text{best}} := f_{\text{new}}$	
$\hat{\mathbf{y}}_n := \hat{\mathbf{y}}_n$	
$\mathbf{H}_n :=$ Jacobian of \mathbf{h}_n at $\hat{\mathbf{x}}_n$?
$\mathbf{w}_n := \mathbf{H}'_n \hat{\mathbf{e}}_n$	$N(2ML)$
$\mathbf{\Pi}_n^{-1} := (\mathbf{H}'_n \mathbf{H}_n)$	$N(2M^2L)$
$\mathbf{\Lambda}_n :=$ diagonal elements of $\mathbf{\Pi}_n^{-1}$	
$\lambda := 0.1\lambda$	1
else	
$\lambda := 10\lambda$	1
} until $\Phi_{\boldsymbol{\alpha}}(\hat{\mathbf{x}})$ decreases insignificantly.	
compute CV_0 score for $\hat{\mathbf{x}}_{\boldsymbol{\alpha}}$	$N(\frac{37}{2}M^3)$

Table 9.1: Iterative nonlinear estimation algorithm and computational requirements.

Figure 9.1: Noisy image data for curved edge estimation example.

Figure 9.2: Noisy image data for straight edge estimation example.

9.6 Edge Estimation Application

One simple application of the nonlinear smoothing algorithm described above is to the problem of estimating the position of edges in digital images. Consider Figures 9.1 and 9.2; each of the ($N = 64$) rows of these images contains ($L = 64$) samples of a step function of unknown shift ($M = 1$). If the edge is known to be straight, then high accuracy techniques exist for estimating the edge [160]. However, if the edge is smoothly varying curve, the nonlinear estimation approach of this chapter is applicable.

An approximate model for the measurement function for this problem is:

$$h_i(\tau) \triangleq \int_{i-1}^i \mathbf{1}_{\{s \leq \tau\}} ds, \quad (9.11)$$

where

$$\mathbf{1}_{\{s \leq \tau\}} = \begin{cases} 1, & s \leq \tau \\ 0, & s > \tau \end{cases},$$

with corresponding Jacobian:

$$\frac{\partial h_i(\tau)}{\partial \tau} = \mathbf{1}_{\{|i-\tau-1/2| \leq 1/2\}}.$$

We generated the data displayed in Figures 9.1 and 9.2 by using (9.11) and adding Gaussian noise with variance $\sigma^2 = 0.25$. The resulting SNR ($= 1/\sigma$) is 2.

Assuming that the underlying edge is smoothly varying (which Figures 9.1 and 9.2 do seem to suggest), the only remaining requirement for the nonlinear smoothing algorithm is to provide an initial estimate. We used the following simple heuristic: a temporary copy of each row of the image was convolved with an approximate matched filter kernel $[1, 1, 1, 1, 1, 1, 0, -1, -1, -1, -1, -1, -1]$, and the index of the pixel with maximum value was stored. This set of $N = 64$ numbers was then median filtered, and the result was the initial estimate of the edge position.

We do not have any reason to stipulate a particular smoothing parameter, so we use cross-validation. To verify the CV_0 approximation, we show in Figure 9.3 a plot of the mean-squared error and the CV_0 score as a function of α for the data set shown in Figure 9.1, where

$$\text{MSE}(\boldsymbol{\alpha}) \triangleq \frac{1}{N} \sum_{n=1}^N \|\hat{\mathbf{g}}\boldsymbol{\alpha}(t_n) - \mathbf{g}(n)\|^2.$$

The minimum of the CV_0 curve is very close to the minimum of the MSE curve, thus our approximation for the CV score is useful for achieving accurate estimates. The underlying curve in Figure 9.2 is truly a straight line. Hence, as shown in Figure 9.4, the MSE is monotonically decreasing with increasing α . Because of the low signal to noise ratio, the CV_0 score decreases to a certain point and then increases again. Nevertheless, the minimum of CV_0 does occur where the MSE is reasonably small.

Figures 9.5 and 9.6 show a comparison of the true and the estimated position functions for the optimal α 's. The algorithm adapted itself to both the curved edge and the straight edge—choosing a much larger value

Figure 9.3: Comparison of MSE and CV_0 for curved edge example.

for the smoothing parameter in the latter case. This example highlights the versatility of this nonparametric paradigm. Figure 9.7 shows plots of the estimation errors for the above examples. The subpixel errors demonstrate the estimation accuracy of this approach.

Figure 9.4: Comparison of MSE and CV_0 for straight edge example.

Figure 9.5: True (solid) and estimated (dashed) edge position from Figure 9.1.

Figure 9.6: True (solid) and estimated (dashed) edge position from Figure 9.2.

Figure 9.7: Estimation errors for edge estimation examples.

9.7 Summary

We have presented an iterative algorithm for nonlinear estimation of a smooth vector-valued function, based on a nonparametric optimality criterion. This algorithm provides an alternative to the EKF that is useful for off-line processing. We have suggested one approximate method for choosing the smoothing parameter automatically. There are a plethora of methods in use for the linear case, including robust choices [150]; a detailed comparison of these methods in the nonlinear case is an open problem.

That our algorithm requires an initial estimate for every state is a mixed blessing. Recursive formulae have also been developed for linear spline smoothing [161, 138]. Perhaps an extension of this work would yield a recursive nonlinear smoother that would only require a single initial state.

In this chapter, we have demonstrated the potential of this algorithm on a simple edge-estimation problem. In addition to the 3-D reconstruction problem of this thesis, other potential applications include biomechanics [139] (tracking the movement of limbs from photographic images), and geophysics [162] (estimating continental plate motion from surface measurements).

Chapter 10

Discussion

10.1 Summary of Contributions

This thesis has described a new object-based method for reconstructing 3-D descriptions of arterial trees from a few projections. The method captures our *a priori* knowledge of the structure of arterial trees in a parametric object model, and quantifies arterial smoothness using new nonparametric smoothing algorithms. By incorporating this *a priori* knowledge into an optimality criterion, we have translated the reconstruction problem into a parameter estimation problem. We developed and implemented an estimation algorithm tailored to this problem, and demonstrated subpixel accuracy reconstructions from as few as four noisy projection images.

We have generalized the measurement model to account for the time-variations of contrast density; an essential extension for MR angiography. We have generalized the object model to accommodate branching arteries. These generalizations allowed us to break through the ‘simulated data’ barrier, and we demonstrated the first *in vivo* reconstructions of an arterial tree with an object-based method. We have also demonstrated the robustness and versatility of the elliptical model by applying it to simulated projections of crescent-shaped cross-sections.

Essential to these low-SNR reconstructions was our use of the smoothness properties of arteries. We generalized the linear, scalar spline smoothing technique to nonlinear, vector measurements. We also generalized the method of cross-validation to these cases. The nonparametric smoothing algorithms are very practical for natural scenes such as angiograms, since the difficult task of deciding how much to smooth is addressed automatically.

The promise of this method is perhaps best demonstrated by Figures 7.3 and 7.4, which show that accurate reconstruction of bifurcations is achievable with parametric models. Note that an attempt to reconstruct intersecting ellipses on a local, slice-by-slice basis would be too sensitive to noise; it is the powerful *a*

priori knowledge of smoothness that makes our global approach effective. Figures 4.7 and 4.8 are also very encouraging results; they show that accurate estimation of the arterial radius is achievable even without assuming that the radius function is smooth, provided that the position and density functions are smooth.

As noted by Ross *et al.* [3]: “Some digital techniques are in use in clinical practice, but application of these promising approaches is not yet widespread.” The author hopes that by having addressed some of the limitations of the previous methods, this dissertation will be a step towards a clinically useful method. Unfortunately, a possible disadvantage of this approach is its complexity; our implementation consists of over ten thousand lines of C programs. However, computer capabilities have risen while the prices have dropped, and our reconstruction times are reasonable (a few minutes) on an affordable workstation. The emphasis on reducing computations that pervaded earlier work in quantitative angiography is unjustifiable now, and researchers will be able to address the remaining challenges by considering increasingly sophisticated models.

10.2 Open Problems

The theory we present has the potential of providing a fully automatic reconstruction algorithm. However, like many methods, the current implementation of our algorithm requires some manual initialization. Automating this will be a necessary step towards making the algorithm useful clinically. One possible approach would use a detection algorithm based on the outer two minimizations of (6.4). Brute force minimization of (6.4) would be impractical computationally; one will need to exploit the structure of arterial trees as done in Chapter 6. Automating the procedure should be relatively easier in the high-SNR case, as with intra-arterial contrast studies [58].

The most important areas for future work are implementing and demonstrating the cone-beam reconstruction algorithm for X-ray angiography, and extending the models to accommodate multi-valued generalized cylinders. A more extensive analysis of the constraints under which the approximations used to develop the cone-beam algorithm is also needed. When addressing the multi-valued problem it should be useful to consider the paradigm that has led to the single-valued object reconstruction algorithm: first consider a single object in a single view, then a single object in multiple views, and perhaps apply the AM iterations to accommodate multiple branching objects. The author suspects it will be more fruitful to first consider the multi-valued problem in a more general projection geometry than the cylindrical one considered here. For example, since three ideal projections are sufficient for reconstructing an ellipse, it should be possible to reconstruct a multi-valued object from projections in six directions: the three coordinate axes and the three bisectors of pairs of those axes.

In addition, there remain a wealth of unanswered questions pertaining to 3-D reconstruction. How should one best choose the smoothing factor α for arterial trees? Should α be the same for every object, or should α scale with object size since smaller arteries tend to be more tortuous? In MR, increasing the resolution by

decreasing the field-of-view (FOV) causes a decrease in signal energy. What is the optimal FOV for a given anatomical region? How large a rotation angle (θ_r) is required for a dual bi-plane projection geometry for a given desired accuracy?

We have demonstrated single-object reconstruction with a smooth background from a simulated projection. The accuracy of such an approach needs to be examined more carefully on real X-ray angiograms. In intra-arterial angiography, the typical signal-to-noise ratio is quite high, but the signal-to-background ratio may be poor. The regular characteristics of arterial projections suggest that morphological filters should be useful for reducing background interference.

Although we have designed a method that accounts for statistical measurement noise, there is another source of error that needs further attention. Most X-ray systems are not positionable precisely, so the 3-D coordinate system for each projection must be calibrated indirectly, typically from projections of a phantom. What is the sensitivity of a parametric reconstruction algorithm to errors in this calibration? The author suspects that the multiple-view methods that use the fewest views are the most vulnerable to such errors, since small errors might tend to ‘average out’ when dozens of views are used. However, calibration errors are similar in character to the errors introduced by the parallel approximation to a cone-beam geometry, so in light of the results of Section 7.5, perhaps the sensitivity is reasonable. For reconstruction from MR angiograms, the consequences of vessels of different phase being superimposed in a projection needs further examination. This may be challenging to study since plastic phantoms induce susceptibility artifacts into MR images.

In our object-based approach, the ellipse area is not a parameter, but it can be computed directly (by $\pi\hat{r}^2$) once the radius is estimated. For arterial segments with overlap-free projections, one could also compute the density-weighted area. Deviations from the elliptical model could then be tested by applying a generalized likelihood-ratio test that compares the densitometric area with $\hat{\rho}\pi\hat{r}^2$. The p-value of the deviations could be reported graphically¹ to indicate potential non-elliptical lesions to the physician. It would also be useful to report confidence intervals for the parameter estimates to the physician. Unfortunately, the theory of confidence intervals for nonparametric smoothing is not developed firmly. One could certainly report the Cramer-Rao lower bound, but we frequently exceed that bound by exploiting smoothness. One approach may be to ‘simulate the posterior’ by generating synthetic projections of the estimated arterial tree with comparable SNR, re-estimating the arterial tree from the synthetic projections, and then looking at the variations over several noise realizations. This may be less time-consuming than one might expect, since one could use the originally estimated arterial tree to initialize the iterative algorithm at a point reasonably close to the optimal estimate.

Our approach has been to avoid enforcing plausible constraints such as equality of the density of overlapping ellipses. Though such constraints could reduce the degrees of freedom and thereby decrease the estimate variance, they could also increase potential modeling error. Nevertheless, if the SNR is very low, it may be

¹At the risk of being ridiculed, one might suggest using color for this purpose.

necessary to use even more *a priori* knowledge. The branching-spline method of Silverman [140], applied to agricultural data originally, may be a one useful approach to enforcing tighter constraints between objects that branch.

As mentioned in Chapters 8 and 9, several research opportunities also remain in nonparametric smoothing. In particular, the cross-validation score for nonlinear measurements is an approximation that needs to be evaluated more carefully. We have based our smoothness penalties on a squared second-derivative criterion. This is certainly appropriate for the position parameters, but for the other parameters a squared first-derivative may be more appropriate. This is true especially for the ellipse orientation parameter. A careful examination of Figure 7.24 reveals a ‘spiral’ character to some of the smaller objects. This is due to a linear orientation function that is completely unpenalized by the second-derivative, even though intuition tells us that such an object is less likely than a non-spiral object. The nonparametric smoothing algorithm and software could be easily generalized to accommodate different penalty functions for the different parameters.

As the above theoretical issues are resolved, increasing attention should be paid to improving the user-interface software, if the method has hopes of being used clinically. Fortunately, the arterial tree descriptions in an object-based approach are very amenable to user interaction. Shaded-surface displays can be generated very rapidly at any projection angle, particular segments of the tree can be identified easily for closer examination, and hemodynamic factors can be computed directly from the parametric description.

One of the most compelling motivations for using as few projections as possible is the practical difficulty in acquiring more than a few ‘simultaneous’ projections. However, perhaps simultaneous projections are unnecessary for arterial reconstruction, since arterial tree motion is constrained. Wu [63] made effective use of multiple cine-projections for simple position and area estimates; it would be interesting to incorporate that approach into the global estimation framework of this thesis to utilize the 4-D information available from cine-projections more completely.

Bibliography

- [1] K. L. Gould. Percent coronary stenosis: battered gold standard or clinical practicality? *Journal of American College of Cardiology*, 11(4):886–888, April 1988.
- [2] D. J. Skorton, M. L. Marcus, P. T. Kirchner, and S. M. Collins. Current needs and future directions in cardiac imaging and image processing. In *Applications of Digital Computer Techniques to Cardiac Imaging*, chapter 16, pages 419–425. McGraw Hill, B.C., 1986.
- [3] J. Ross, C. Fisch, et al. Guidelines for coronary angiography. *Circulation*, 76(4):963A–977A, October 1987.
- [4] B. G. Brown, E. Bolson, M. Frimer, and H. T. Dodge. Quantitative coronary arteriography. *Circulation*, 55(2):329–337, February 1977.
- [5] K. L. Gould. Quantification of coronary artery stenosis in vivo. *Circulation Research*, 57(3):341–353, September 1985.
- [6] S. Paulin. Assessing the severity of coronary lesions with angiography. *New England Journal of Medicine*, 316(22):1405–1407, May 1987.
- [7] S. E. Epstein, A. A. Quyyumi, and R. O. Bonow. Sudden cardiac death without warning. *New England Journal of Medicine*, 321(5):320–324, August 1989.
- [8] S. Glagov, E. Weisenberg, C. K. Zarins, R. Stankunavicius, and G. J. Kolettis. Compensatory enlargement of human atherosclerotic coronary arteries. *New England Journal of Medicine*, 316(22):1371–1375, 1987.
- [9] G. Jividen and Rangachar Kasturi. *A Survey of Image Processing Techniques for Angiograms*, volume TR-88-048. Penn. State, University Park Campus, 1988.
- [10] K. Shmueli. *Estimation of blood vessel boundaries in X-ray images*. PhD thesis, Stanford University, Stanford, CA., August 1981.

- [11] J. H. C. Reiber, C. J. Kooijman, C. J. Slager, J. J. Gerbrands, J. C. H. Schuurbiens, A. Den Boer, W. Wijns, P. W. Serruys, and P. G. Hugenholtz. Coronary artery dimensions from cineangiograms—methodology and validation of a computer-assisted analysis procedure. *IEEE Transactions on Medical Imaging*, 3(3):131–141, September 1984.
- [12] T. N. Pappas and J. S. Lim. A new method for estimation of coronary artery dimensions in angiograms. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 36(9):1501–1513, September 1988.
- [13] S. R. Fleagle, M. R. Johnson, C. J. Wilbricht, D. J. Skorton, R. F. Wilson, C. W. White, M. L. Marcus, and S. M. Collins. Automated analysis of coronary arterial morphology in cineangiograms: geometric and physiologic validation in humans. *IEEE Transactions on Medical Imaging*, 8(4):387–400, December 1989.
- [14] K. Kitamura, J. M. Tobis, and J. Sklansky. Estimating the 3-D skeletons and transverse areas of coronary arteries from biplane angiograms. *IEEE Transactions on Medical Imaging*, 7(3):173–187, September 1988.
- [15] Y. Bresler. *Model Based Estimation Techniques for 3-D Reconstruction from Projections*. PhD thesis, Stanford University, Stanford, CA., December 1985.
- [16] R. A. Kruger, D. R. Reinecke, S. W. Smith, and R. Ning. Reconstruction of blood vessels from X-ray subtraction projections: Limited angle geometry. *Medical Physics*, 14(6):940–948, November 1987.
- [17] J. Liu, D. Nishimura, and A. Macovski. Generalized tomosynthesis for focusing on an arbitrary surface. *IEEE Transactions on Medical Imaging*, 8(2):168–172, June 1989.
- [18] C. Dumoulin et al. Three-dimensional phase contrast angiography. *Magnetic Resonance in Medicine*, 9(1):139–149, January 1989.
- [19] T. Masaryk et al. 3DFT MR angiography of the carotid bifurcation: A comparison of time-of-flight techniques. In *Proc. Eighth SMRM*, volume 2, page 164, August 1989.
- [20] M. Block, A. A. Bove, and E. L. Ritman. Coronary angiographic examination with the dynamic spatial reconstructor. *Circulation*, 70(2):209–216, August 1984.
- [21] W. E. Higgins, W. J. T. Spyra, and E. L. Ritman. Automatic extraction of the arterial tree from 3-D angiograms. In *Proceedings, 11th Annual Intl. Conf. of the IEEE Engr. in Medicine and Biology Society*, pages 563–564, November 1989.
- [22] S. Lobregt, P. W. Verbeek, and F. C. A. Groen. Three-dimensional skeletonization: Principle and algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(1):75–77, January 1980.

- [23] J. A. K. Blockland, A. M. Vossepoel, A. R. Bakker, and E. K. J. Pauwels. Delineating elliptical objects with an application to cardiac scintigrams. *IEEE Transactions on Medical Imaging*, 6(1):57–66, March 1987.
- [24] W. Lin, C. Liang, and C. Chen. Dynamic elastic interpolation for 3-D medical image reconstruction from serial cross sections. *IEEE Transactions on Medical Imaging*, 7(3):225–232, September 1988.
- [25] D. G. Nishimura, A. Macovski, J. M. Pauly, and S. M. Conolly. MR angiography by selective inversion recovery. *Magnetic Resonance in Medicine*, 4:193–202, 1987.
- [26] S Wang et al. Improved static tissue suppression for selective inversion recovery angiography, 1990. In preparation.
- [27] K. Barth, U. Faust, A. Both, and K. Wedekind. A critical examination of angiographic quantitation by digital image processing. In *Proc. IEEE Comp. Soc. Intl. Symp. on Medical Imaging and Image Interpretation*, pages 71–76. IEEE Press, October 1982.
- [28] C. J. Kooijman, J. H. C. Reiber, J. J. Gerbrands, J. C. H. Schuurbiens, C. J. Slager, A. Den Boer, and P. W. Serruys. Computer-aided quantitation of the severity of coronary obstruction from single view cineangiograms. In *Proc. IEEE Comp. Soc. Intl. Symp. on Medical Imaging and Image Interpretation*, pages 59–63. IEEE Press, October 1982.
- [29] J. R. Spears, T. Sandor, A. V. Als, M. Malagold, J. E. Markis, W. Grossman, J. R. Serur, and S. Paulin. Computerized image analysis for quantitative measurement of vessel diameter from cineangiograms. *Circulation*, 68(2):453–461, August 1983.
- [30] T. N. Pappas and J. S. Lim. Estimation of coronary artery boundaries in angiograms. In *Proc. SPIE 504, Applications of Digital Image Processing VII*, pages 312–320, 1984.
- [31] W. Pratt. *Digital Image Processing*. John Wiley and Sons, New York, 1978.
- [32] J. Modestino and R. Fries. Edge detection in noisy images using recursive digital filtering. *Comp. Graphics Image Proc.*, 6, 1977.
- [33] G. Ashkar and J. Modestino. The contour extraction problem with biomedical applications. *Comp. Graphics Image Proc.*, 7, 1978.
- [34] N. Hahi and S. Lopez-Mora. Estimation-detection of object boundaries in noisy images. *IEEE Transactions on Automatic Control*, 23(5), October 1978.
- [35] T. N. Pappas. *Estimation of Coronary Artery Dimensions from Angiograms*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA., April 1987.

- [36] T. Sandor, A. D'Adamo, W. B. Hanlon, and J. R. Spears. High precision quantitative angiography. *IEEE Transactions on Medical Imaging*, 6(3):258–265, September 1987.
- [37] D. O'Handley et al. Picture analysis applied to biomedicine. *Comp. Graphics Image Proc.*, 2, 1973.
- [38] K. Castleman et al. Vessel edge detection in angiograms: An application of the Wiener filter. In *Digital Signal Processing*. Point Lobos Press, N. Hollywood, CA, 1980.
- [39] D. M. Weber. Absolute diameter measurements of coronary arteries bases on the first zero crossing of the Fourier spectrum. *Medical Physics*, 16(2):188–196, March 1989.
- [40] K. Shmueli, W. R. Brody, and A. Macovski. Estimation of blood vessel boundaries in X-ray images. In *Proc. SPIE 314, Digital Radiography*, pages 279–286, Stanford, CA, September 1981.
- [41] K. Shmueli, W. R. Brody, and A. Macovski. Estimation of blood vessel boundaries in X-ray images. *Optical Engineering*, 22(1):110–116, January 1983.
- [42] R. A. Kruger. Estimation of the diameter of and iodine concentration within blood vessels using digital radiography devices. *Medical Physics*, 8(5):652–658, September 1981.
- [43] M. Siebes, D. Z. D'Argenio, and R. H. Selzer. Computer assessment of hemodynamic severity of coronary artery stenosis from angiograms. *Computer Methods and Programs in Biomedicine*, 21:143–152, 1985.
- [44] D. G. Nishimura. *Signal Processing Methods for Intravenous Angiography*. PhD thesis, Stanford University, Stanford, CA., December 1983.
- [45] K. R. Hoffmann, K. Doi, H. Chan, L. Fencial, H. Fujita, and A. Muraki. Automated tracking of the vascular tree in DSA images using a double-square-box region-of-search algorithm. In *Proc. SPIE 626*, pages 326–333, 1986.
- [46] Y. Sun. Automated identification of vessel contours in coronary arteriograms by an adaptive tracking algorithm. *IEEE Transactions on Medical Imaging*, 8(1):78–88, March 1989.
- [47] T. V. Nguyen and J. Sklansky. A fast skeleton-finder for coronary arteries. In *Proc. 8th IEEE Intl. Conf. on Pattern Recognition*, pages 481–483, 1986.
- [48] T. V. Nguyen and J. Sklansky. Computing the skeleton of coronary arteries in cineangiograms. *Computers and Biomedical Research*, 19:428–444, 1986.
- [49] S. A. Stansfield. ANGY: a rule-based expert system for automatic segmentation of coronary vessels from digital subtracted angiograms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(2):188–199, March 1986.

- [50] D. Ergener, S. Gunasekaran, M. Trefler, and T. Y. Young. Sequential analysis of angiograms by an expert system. In *Proc. SPIE 914, Medical Imaging II: Image Formation, Detection, Processing, and Interpretation*, pages 613–619, 1988.
- [51] P. H. Eichel, E. J. Delp, K. Koral, and A. J. Buda. A method for a fully automatic definition of coronary arterial edges from cineangiograms. *IEEE Transactions on Medical Imaging*, 7(4):313–320, December 1988.
- [52] G. Besson. Vascular segmentation using snake transforms and region growing. In *Proc. SPIE 1092, Medical Imaging III: Image Processing*, pages 429–435, 1989.
- [53] S. Chaudhuri, S. Chatterjee, N. Katz, M. Nelson, and M. Goldbaum. Detection of blood vessels in retinal images using two-dimensional matched filters. *IEEE Transactions on Medical Imaging*, 8(3):263–269, September 1989.
- [54] S. Chang and C. K. Chow. The reconstruction of three-dimensional objects from two orthogonal projections and its application to cardiac cineangiography. *IEEE Transactions on Computers*, 22(1):18–28, January 1973.
- [55] J. J. Gerbrands, G. J. Troost, J. H. C. Reiber, C. J. Kooijman, and C. H. Slump. 3-D reconstruction of coronary arterial segments from two projections. In *Proc. 6th IEEE Intl. Conf. on Pattern Recognition*, pages 270–272, Munich, 1982. IEEE.
- [56] J. H. C. Reiber, J. J. Gerbrands, G. J. Troost, C. J. Kooijman, and C. H. Slump. 3-D reconstruction of coronary arterial segments from two projections. In P. Heintzen and R. Breanecke, editors, *Digital Imaging in Cardiovascular Radiology*. G. Thieme Verlag, New York, 1983.
- [57] P. Fenster, J. Barba, M. Surdiaz, and K. K. Wong. Arterial cross-section reconstruction from bi-plane X-ray shadowgraphs. In *Proc. SPIE 767, Medical Imaging*, pages 433–440, 1987.
- [58] P. Suetens, P. J. Jansen, A. Haegemans, A. Oosterlinck, and J. Gybels. 3-D reconstruction of the blood vessels of the brain from a stereoscopic pair of subtraction angiograms. *Image and Vision Computing*, 1(1):43–51, February 1983.
- [59] J. R. Spears, T. Sandor, D. S. Baim, and S. Paulin. The minimum error in estimating coronary luminal cross-sectional area from cineangiographic diameter measurements. *Catheterization and Cardiovascular Diagnosis*, 9:119–128, 1983.
- [60] M. Siebes, M. Gottwik, and M. Schlepper. Quantitative angiography: Experimental studies on the representation of model coronary arteries in angiographic films. In *Proc. IEEE Comp. Soc. Intl. Symp. on Medical Imaging and Image Interpretation*, pages 65–70, 1982.

- [61] K. Kitamura, J. M. Tobis, and J. Sklansky. Estimating the X-ray intercepted areas and boundaries of coronary artery trees. In *Proc. 8th IEEE Intl. Conf. on Pattern Recognition*, pages 478–480, 1986.
- [62] D. L. Parker, D. L. Pope, R. Van Bree, and H. W. Marshall. Three-dimensional reconstruction of moving arterial beds from digital subtraction angiography. *Comp. and Biomed. Res.*, 20:166–185, 1987.
- [63] J. Wu, D. L. Parker, G. R. Caputo, and H. W. Marshall. A clinical feasibility study of 3D reconstruction and measurement of coronary arteries. *Dynamic Cardiovascular Imaging*, 1(4):273–278, 1988.
- [64] J. J. Gerbrands, J. H. C. Reiber, B. Scholts, G. Langhout, and C. J. Kooijman. Structural analysis of the coronary arterial tree. In *Proc. IEEE Comp. Soc. Intl. Symp. on Medical Imaging and Image Interpretation*, pages 54–58, 1982.
- [65] H. C. Kim, B. G. Min, T. S. Lee, S. J. Lee, C. W. Lee, J. H. Park, and M. C. Han. Three-dimensional digital subtraction angiography. *IEEE Transactions on Medical Imaging*, 1(2):152–158, October 1982.
- [66] K. R. Hoffmann, K. Doi, H. Chan, and K. Chua. Computer reproduction of the vasculature using an automated tracking method. In *Proc. SPIE 767, Medical Imaging*, pages 449–453, 1987.
- [67] C. Venaille, D. Mischler, R. Collorec, J. Y. Catros, and J. L. Coatrieux. Automatic 3D reconstruction of vascular networks from three projections: a simulated annealing approach. In *Proceedings, 11th Annual Intl. Conf. of the IEEE Engr. in Medicine and Biology Society*, pages 565–566, Seattle, November 1989.
- [68] X. Jiye and O. H. Kapp. Three dimensional algebraic reconstruction from projections of multiple grey level objects. *Optik*, 72(3):87–94, 1986.
- [69] J. R. Spears, T. Sandor, R. Kruger, W. Hanlon, S. Paulin, and G. Minerbo. Computer reconstruction of luminal cross-sectional shape from multiple cineangiographic views. *IEEE Transactions on Medical Imaging*, 2(1):49–54, March 1983.
- [70] H. Haneishi, T. Masuda, N. Ohyama, and T. Honda. Three-dimensional blood vessel reconstruction by simulated annealing. *Optics Letters*, 14(20):1095–1097, October 1989.
- [71] A. Rougee, K. M. Hanson, and D. Saint-Felix. Comparison of 3-D tomographic algorithms for vascular reconstruction. In *Proc. SPIE 914, Medical Imaging II: Image Formation, Detection, Processing, and Interpretation*, pages 397–405, 1988.
- [72] Y. Troussset, D. Saint-Felix, A. Rougée, and C. Chardenon. Multiscale cone-beam X-ray reconstruction. In *Proc. SPIE Medical Imaging IV*, February 1990.

- [73] D. J. Rossi and A. S. Willsky. Reconstruction from projections based on detection and estimation of objects—parts I & II: Performance analysis and robustness analysis. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 32(4):886–906, August 1984.
- [74] D. J. Rossi. *Reconstruction from Projections Based on Detection and Estimation of Objects*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA., September 1982.
- [75] D. Rossi and A. Willsky. Reconstruction from projections based on detection and estimation of objects. In *Proc. IEEE Conf. on Acoustics, Speech, and Signal Processing*, pages 128–130, 1983.
- [76] D. J. Rossi, A. S. Willsky, and D. M. Spielman. Object shape estimation from tomographic measurements—a performance evaluation. *Signal Processing*, 18(1):63–88, September 1989.
- [77] Y. Bresler and A. Macovski. 3-D reconstruction from projections based on dynamic object models. In *Proc. IEEE Conf. on Acoustics, Speech, and Signal Processing*, San Diego, CA, March 1984.
- [78] Y. Bresler and A. Macovski. Three-dimensional reconstruction from projections with incomplete and noisy data by object estimation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35(8):1139–1152, August 1987.
- [79] Y. Bresler and A. Macovski. Estimation of 3-D shape of blood vessels from X-ray images. In *Proc. IEEE Intl. Symp. Medical Images and Icons*, Arlington, VA, July 1984.
- [80] Y. Bresler and A. Macovski. 3-D reconstruction from projections with incomplete and noisy data by object estimation. In *Proc. Intl. Conf. on Computers, Systems, and Signal Processing*, Bangalore, India, December 1984.
- [81] Y. Bresler and A. Macovski. A hierarchical Bayesian approach to reconstruction from projections of a multiple object 3-D scene. In *Proc. 7th IEEE Intl. Conf. on Pattern Recognition*, Montreal, Canada, July 1984.
- [82] Y. Bresler, J. A. Fessler, and A. Macovski. A Bayesian approach to reconstruction from incomplete projections of a multiple object 3-D domain. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(8):840–858, August 1989.
- [83] Y. Bresler, J. A. Fessler, and A. Macovski. Model based estimation techniques for 3-D reconstruction from projections. *Machine Vision and Applications*, 1(2):115–126, 1988.
- [84] Y. Bresler. Paraphrased from a personal electronic mail message, 1989.
- [85] R. McEliece. *The Theory of Information and Coding*, volume 3 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, New York, 1984.

- [86] A. Macovski. *Medical Imaging Systems*. Prentice-Hall, New Jersey, 1983.
- [87] J. A. Fessler and A. Macovski. 3-D reconstruction of vessels with stenoses and aneurysms from dual bi-plane angiograms. In *Proc. SPIE 1092, Medical Imaging III: Image Processing*, pages 22–32, February 1989.
- [88] M. A. Simons and R. A. Kruger. Vessel diameter measurement using digital subtraction radiography. *Investigative Radiology*, 20(4):510–516, July 1985.
- [89] M. A. Simons, R. A. Kruger, and R. L. Power. Cross-sectional area measurements by digital subtraction videodensitometry. Presented at 33rd Meeting of Univ. Radiologists, May 1985.
- [90] D. Pickens and J. Fitzpatrick. Phantom design to evaluate a three-dimensional motion correction algorithm in DSA of the coronary arteries. In *Proc. SPIE 914, Medical Imaging II: Image Formation, Detection, Processing, and Interpretation*, pages 707–714, 1988.
- [91] L. Tran and J. Sklansky. Subpixel flexible registration for quantitative angiography. In *Proc. Computers and Radiology Conf.*, June 1989.
- [92] O. Nalcioglu, W. W. Roeck, T. Reese, L. Z. Qu, J. M. Tobis, and W. L. Henry. Background subtraction algorithms for videodensitometric quantification of coronary stenosis. *Machine Vision and Applications*, 1:155–162, 1988.
- [93] D. G. Nishimura, A. Macovski, J. I. Jackson, R. S. Hu, C. A. Stevick, and L. Axel. Magnetic resonance angiography by selective inversion recovery using a compact gradient echo sequence. *Magnetic Resonance in Medicine*, 8:96–103, 1988.
- [94] Steve Webb. *From the watching of shadows: the origins of radiological tomography*. A. Hilger, Bristol, 1990.
- [95] D. G. Nishimura, J. M. Pauly, and A. Macovski. MR angiography using projection-reconstruction imaging. In *Magnetic Resonance Imaging - SMRI 8th Annual Mtg. Program & Abstracts*, volume 8 of 344, page 107, February 1990.
- [96] D G Nishimura, J M Pauly, and A Macovski. MR angiography using projection-reconstruction imaging. In *Magnetic Resonance Imaging - SMRI 8th Annual Mtg Program & Abstracts*, volume 8 of 344, page 107, 1990.
- [97] A. Macovski and D. C. Noll. Homodyne detection for improved SNR and partial k-space reconstructions. In *Book of Abstracts*, volume 2, page 815. Soc. of Magnetic Resonance in Medicine, August 1988.

- [98] D C Noll, D G Nishimura, and A Macovski. Homodyne detection in magnetic resonance imaging. *IEEE Tr. Med. Im.*, 10(2):154–163, June 1991.
- [99] W. A. Edelstein, G. H. Glover, C. J. Hardy, and R. W. Redington. The intrinsic signal-to-noise ratio in NMR imaging. *Magnetic Resonance in Medicine*, 3:606–618, 1986.
- [100] D. P. Giddens, C. K. Zarins, S. Glagov, B. K. Bharadvaj, and D. N. Ku. Flow and atherogenesis in the human carotid bifurcation. In G. Schettler et al., editors, *Fluid Dynamics as a Localizing Factor for Atherosclerosis*, pages 38–45. Springer-Verlag, Berlin, 1983.
- [101] P. Grottum, A. Svinland, and L. Walløe. Localization of atherosclerotic lesions in the bifurcations of the main left coronary artery. *Atherosclerosis*, 47:55–62, 1983.
- [102] T. Binford. Visual perception by computer. In *Proc. IEEE Conf. on Systems and Controls*, Miami, Florida, December 1971.
- [103] G. Agin and T. Binford. Computer description of curved objects. *IEEE Transactions on Computers*, C-25:439–449, April 1976.
- [104] D Marr. *Vision*. Freeman, San Francisco, 1982.
- [105] J. A. Fessler, D. G. Nishimura, and A. Macovski. Model-based 3-D reconstruction of branching vessels. In *Proceedings, 11th Annual Intl. Conf. of the IEEE Engr. in Medicine and Biology Society*, pages 561–562, Seattle, November 1989.
- [106] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C*. Cambridge Univ. Press, 1988.
- [107] M. Zakai and J. Ziv. On the threshold effect in radar range estimation. *IEEE Transactions on Information Theory*, pages 167–169, January 1969.
- [108] B W Silverman. Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *J. Royal Stat. Soc. Ser. B*, 47(1):1–52, 1985.
- [109] M. Bertero, T. Poggio, and V. Torre. Ill posed problems in early vision. *Proceedings of the IEEE*, 76(8):869–899, August 1988.
- [110] G Wahba and S Wold. A completely Automatic French curve: fitting spline functions by cross validation. *Comm. in Statistics*, 4(1):1–17, 1975.
- [111] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society Series B*, 39:1–38, 1977.

- [112] M. Feder and E. Weinstein. Parameter estimation of superimposed signals using the EM algorithm. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 36(4):477–489, April 1988.
- [113] I. Ziskind and M. Wax. Maximum likelihood localization of multiple sources by alternating projection. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 36(10):1553–1560, October 1988.
- [114] M. Lichtenstein and T. Young. The resolution of closely spaced signals. *IEEE Transactions on Information Theory*, 14(2):288–293, March 1968.
- [115] R. DeFigueiredo and Chia-Ling Hu. Waveform feature extraction based on Tauberian approximation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(2):105–116, March 1982.
- [116] R. DeFigueiredo and Andreas Gerber. Separation of superimposed signals by a cross-correlation method. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 31(5):1084–1089, October 1983.
- [117] H. Akaike. Statistical predictor identification. *Ann. Inst. Statist. Math.*, 22:203–217, 1970.
- [118] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, December 1974.
- [119] J. Rissanen. Modeling by shortest data description. *Automatica*, 14:465–471, 1978.
- [120] M. Wax and T. Kailath. Detection of signals by information theoretic criteria. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 33(2):387–392, April 1985.
- [121] J. Rissanen. Stochastic complexity and modeling. *The Annals of Statistics*, 14(3):1080–1100, 1986.
- [122] J. Wu. On the convergence properties of the EM algorithm. *The Annals of Statistics*, 11(1):95–103, 1983.
- [123] H. Kwakernaak. Estimation of pulse heights and arrival times. *Automatica*, 16:367–377, 1980.
- [124] R. Bracewell. *The Fourier Transform and its Applications*. McGraw-Hill, New York, 1978.
- [125] I J Schoenberg. On interpolation by spline functions and its minimal properties. In *On Approximation Theory, Proc Conf in Oberwolfach, Black Forest*, pages 109–129, 1963.
- [126] C H Reinsch. Smoothing by spline functions. *Numerische Mathematik*, 10:177–183, 1967.
- [127] R L Eubank. *Spline smoothing and nonparametric regression*. Marcel Dekker, New York, 1988.
- [128] S. MacKay, R. Sayre, and M. Potel. 3D Galatea: Entry of three-dimensional moving points from multiple perspective views. *Computer Graphics*, 16(3):213–222, July 1982.

- [129] J A Fessler. Nonparametric smoothing of nonlinear measurements. *IEEE Tr. Sig. Proc.*, 39(4):907–13, April 1991.
- [130] D. Gordon and R. Reynolds. Image space shading of 3-dimensional objects. *Computer Vision, Graphics, and Image Processing*, 29:361–376, 1985.
- [131] J A Fessler. Nonparametric fixed-interval smoothing with vector splines. *IEEE Tr. Sig. Proc.*, 39(24):852–9, April 1991.
- [132] B. D. O. Anderson and J. B. Moore. *Optimal Filtering*. Prentice-Hall, New Jersey, 1979.
- [133] L Ljung. *System identification, theory for the user*. Prentice-Hall, New Jersey, 1987.
- [134] J J Miller and E J Wegman. Vector function estimation using splines. *Journal of Statistical Planning and Inference*, 17:173–180, 1987.
- [135] E J Wegman. Vector splines and the estimation of filter functions. *Technometrics*, 23(1):83–89, February 1981.
- [136] H J Woltring. A Fortran package for generalized cross-validated spline smoothing and differentiation. *Advances in Engineering Software*, 8(2):104–112, 1986.
- [137] H J Woltring, A de Lange, J M G Kauer, and R Huiskes. Instantaneous helical axis estimation via natural cross-validated splines. In A Rohlmann G Bergmann, R Kolbel, editor, *Biomechanics: basic and applied research*, pages 121–128. European Society of Biomechanics, 1986.
- [138] G S Sidhu and H L Weinert. Vector-Valued L_g -Splines II, Smoothing splines. *Journal of Mathematical Analysis and Applications*, 101(2):380–396, July 1984.
- [139] H. J. Woltring. On optimal smoothing and derivative estimation from noisy displacement data in biomechanics. *Human Movement Science*, 4(3):229–230, 1985.
- [140] B W Silverman and J T Wood. The nonparametric estimation of branching curves. *J. Am. Stat. Ass.*, 82(398):551–558, June 1987.
- [141] G Wahba and J Wendelberger. Some new mathematical methods for variational objective analysis using splines and cross validation. *Monthly Weather Review*, 108:1122–1143, 1980.
- [142] D Lee and T Pavlidis. One-dimensional regularization with discontinuities. *IEEE Tr. Patt. Anal. Mach. Int.*, 10(6):822–829, November 1988.
- [143] J. J. Dongarra and E. Grosse. Distribution of mathematical software via electronic mail. *Comm. ACM*, 30:403–407, October 1987.

- [144] C H Reinsch. Smoothing by spline functions II. *Numerische Mathematik*, 16:451–454, 1971.
- [145] H. J. Woltring. A Fortran package for generalized cross-validatory spline smoothing and differentiation. *Advances in Engineering Software*, 8(2):104–112, 1986.
- [146] Carl de Boor. *A practical guide to splines*. Springer Verlag, New York, 1978.
- [147] G. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins Univ. Press, Baltimore, MD., 1983.
- [148] T Kailath. *Linear systems*. Prentice-Hall, New Jersey, 1980.
- [149] P Craven and G Wahba. Smoothing noisy data with spline functions. *Numerische Mathematik*, 31:377–403, 1979.
- [150] T Robinson and R Moyeed. Making robust the cross-validatory choice of smoothing parameter in spline smoothing regression. *Commun Statist.—Theory Meth.*, 18(2):523–539, 1989.
- [151] M F Hutchinson and F R de Hoog. Smoothing noisy data with spline functions. *Numerische Mathematik*, 47:99–106, 1985.
- [152] F R de Hoog and M F Hutchinson. An efficient method for calculating smoothing splines using orthogonal transformations. *Numerische Mathematik*, 50:311–319, 1987.
- [153] G Wahba and S Wold. Periodic splines for spectral density estimation: the use of cross validation for determining the degree of smoothing. *Comm. in Statistics*, 4(2):125–141, 1975.
- [154] G Wahba. Improper priors, spline smoothing, and the problem of guarding against model errors in regression. *J. Royal Stat. Soc. Ser. B*, 40(3):364–372, 1978.
- [155] M J Buckley, G K Eagleson, and B W Silverman. The estimation of residual variance in nonparametric regression. *Biometrika*, 75(2):189–199, 1988.
- [156] P. Kaminski, A. Bryson, and S. Schmidt. Discrete square root filtering: A survey of current techniques. *IEEE Transactions on Automatic Control*, 16(6):727–735, December 1971.
- [157] F O’Sullivan, B S Yandell, and W J Raynor. Automatic smoothing of regression functions in generalized linear models. *J. Am. Stat. Ass.*, 81(393):96–103, March 1986.
- [158] Y. Bresler and A. Macovski. Nonlinear estimation of dynamic shift and shape parameters from set function measurements. In *Automatic Control Conference Proceedings, San Diego, CA*, June 1984.
- [159] D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *J. Soc. Indust. Appl. Math.*, 11(2):431–441, June 1963.

- [160] D. Petkovic, W. Niblack, and M. Flickner. Projection-based high accuracy measurement of straight line edges. *Machine Vision and Applications*, 1:183–199, 1988.
- [161] W E Wecker and C F Ansley. The signal extraction approach to nonlinear regression and spline smoothing. *J. Am. Stat. Ass.*, 78(381):81–89, March 1983.
- [162] M. Matthews. *Stochastic models and methods for inverse problems on space-time domains*. PhD thesis, Stanford University, Stanford, CA., 1990.