USC-SIPI REPORT #193

Recovery of 3-D Motion from 3-D Density Images

by

Samuel Moon-Ho Song

December 1991

Signal and Image Processing Institute
UNIVERSITY OF SOUTHERN CALIFORNIA
Department of Electrical Engineering-Systems
Electrical Engineering Building
University Park/MC-2564
Los Angeles, CA 90089 U.S.A.

Acknowledgements

What seemed like an eternity is finally over. There were many who have helped along the way and they truly have made it all happen. Some have helped in a direct way, lending their technical expertise; and others, just as importantly, have helped indirectly. I have been fortunate to have these people around me and to have had their help through some difficult times. Now, the time has come to express my appreciation and gratitude.

I would like to thank my advisor Dr. R. Leahy, for his meticulous review of this material. His critical comments and suggestions have improved much of the content in this dissertation. Drs. J. Kuo, M. Singh, and C. Synolakis, are gratefully acknowledged for serving on my committee. Their valuable comments have broadened my perspective on this subject. I would like to extend a special thanks to Dr. Kuo whom I have consulted a number of times. His expertise on PDEs and numerical analysis was most invaluable which without, I would have wasted many days and nights.

Dr. B. Brundage of Harbor-UCLA Medical Center was most generous with his time. Despite his busy schedule, he has always made time to meet with me to discuss this subject. He has patiently went through some of the results obtained and gave his expert clinical viewpoint. I am truly grateful for his time and his interest in this material.

The financial support of Hughes Aircraft Company through its fellowship program is gratefully acknowledged. My former and present managers at Hughes were very understanding of my school obligations and have always tried to work things around my fickle schedule. I thank them for creating a pleasant working environment and also for their full support of my scholastic endeavors.

I have had the luck of being part of a family with full of love and caring. My mother, who still enjoys buying clothes for me, has always given her unyielding support throughout my life. I thank her dearly for trusting my judgments. And to my sister Inyoung, who has raised me for a part of my adolescent life, I cannot convey enough appreciation. Her continued love and concern will never be forgotten and I thank her for pushing me to go back to school when I had quit after my master's degree. I also thank my other family members, Inkyung, Inmi and Inae, for their love and support.

I must apologize to my wife Yooha for not having spent enough time together. She was quite understanding of my *normal* working hours—morning till late night. I thank her for her love and much needed encouragements as well as many late-night snacks.

The people of Signal and Image Processing Institute have made this part of my life an intellectually stimulating experience. In particular, I have had close associations with S. G. Kong, K. M. Lee, R. H. Lee, J. Mosher, Z. Wu, X. Yan, and Z. Zhou. I thank them for their friendship which I will long remember. I have especially enjoyed many esoteric discussions with Rae Lee on topics ranging from the theory of Dempster-Shafer reasoning to Korean politics. Thanks are due to J. Mosher who was more than willing to help and to share his vast knowledge on various computer systems. He had helped to put together a MATLAB script file

to display the 3-D vector plots shown in Figure 4.7.

Lastly, but most importantly, I would like to thank my mentor, Dr. Doug Boyd of Imatron Inc., for planting the seed (probably without him knowing) that sparked my interest in medical imaging. He has been a constant source of inspiration. His encouragements and optimism towards this material did wonders in boosting my self-assurance and self-assessment of this work. In many ways, without him, this dissertation would not have materialized.

Contents

Acknowledgements ii

List of Figures vii

List of Tables ix

Abstract x

1 Introduction 1
- 1.1 Motivation and Goals 1
- 1.2 Previous Studies on Cardiac Motion 5
- 1.3 Overview 7
- 1.4 Contributions 8

2 Constraints on the Velocity Field of Moving Media 10
- 2.1 Descriptions of Motion of Deformable Media 11
- 2.2 The Equation of Continuity 12
- 2.3 The Incompressibility Constraint 14
- 2.4 The Divergence-Free Constraint 16

3 Motion Recovery from Density Images: Problem Formulation and Solution Methods 18
- 3.1 Formulation 18
  - 3.1.1 Ill-posed Problems and Regularization Methods 19
  - 3.1.2 Optimization Approach 21
- 3.2 Convexity of Functionals on a Vector Space 23
- 3.3 Existence and Uniqueness of the Solution 26
- 3.4 Solution by Euler-Lagrange Method 30
- 3.5 Boundary Conditions 33
  - 3.5.1 Dirichlet Boundary Condition 33
  - 3.5.2 Natural Boundary Condition 35
- 3.6 Discussion 36

4 Solution in the Discrete Domain: Numerical Methods **38**
4.1 Discretization of the PDEs 39
4.1.1 Dirichlet Boundary Condition 41
4.1.2 Neumann Boundary Condition 41
4.2 Existence and Uniqueness of the Discrete Solution 43
4.2.1 Dirichlet Boundary Condition 43
4.2.2 Neumann Boundary Condition 47
4.2.3 Mixed Boundary Condition 52
4.3 Numerical Algorithms 54
4.3.1 Conjugate Gradient Algorithm 54
4.3.2 Preconditioned Conjugate Gradient Algorithm 56
4.4 Results 61
4.4.1 Simulated Images 61
4.4.2 Cine CT of a beating heart 72
4.5 Preconditioned Conjugate Gradient Algorithm: A Performance Study 78

5 Lagrange Multiplier Method: Obtaining the Regularization Parameters **81**
5.1 Determining One Parameter: Divergence-free Term 82
5.2 Determining Both Parameters 85
5.2.1 Constrained Optimization of Functionals by Dual Space Method 86
5.2.2 Constrained 3-D Motion Recovery Problem 89
5.2.3 Results 92
5.3 Discussion 98

6 Conclusions and Future Research **99**
6.1 Conclusions 99
6.2 Future Research 101

A Euler-Lagrange Equations for Certain Optimization Problems **103**
A.1 Unconstrained Problems 103
A.2 Constrained Problems 107

B Miscellaneous Proofs **111**

C Partial List of Symbols **117**

References **119**

List of Figures

4.1 Simulated images for experiments 1, 2 and 3. (a) Experiment 1: Vertically translating circle, time frames 1 and 2. (b) Experiment 2: Diagonally translating circle, time frames 1 and 2. (c) Experiment 3: Deforming ellipse, time frames 1 and 2. The boundary of the outer region is fixed and the motion arises from the inner regions . 65

4.2 Results of experiment 1 with the boundary outlined. The dotted circle shows the inner circle's position in the second frame. (a) Incompressibility constraint only. (b) Incompressibility and divergence-free constraints. 66

4.3 Results of experiment 2 with the boundary outlined. The dotted circle shows the inner circle's position in the second frame. (a) Incompressibility constraint only. (b) Incompressibility and divergence-free constraints. 67

4.4 Results of experiment 3 with the boundary outlined. The dotted ellipse shows the inner ellipse's position in the second frame. (a) Incompressibility constraint only. (b) Incompressibility and divergence-free constraints. 68

4.5 Results of experiment 2 with different values of γ_1; and $\gamma_2 = 250$. The dotted circle shows the inner circle's position in the second frame. (a) $\gamma_1 = .001$, (b) $\gamma_1 = .005$, (c) $\gamma_1 = .01$, (d) $\gamma_1 = .02$ . . 69

4.6 Simulated images for experiment 4: Vertically translating ellipsoid, time frames 1 and 2. The outer ellipsoid is fixed and the inner ellipsoid translates down one voxel. 70

4.7 Results of experiment 4. 3-D vector field as a function of a 3-D space is projected onto a plane. (a) Incompressibility constraint only. (b) Incompressibility and divergence-free constraints. . . . 71

4.8 Cine CT of a beating heart overlayed with velocity field vectors. Levels 1 through 4, frame 1. 74

4.9 Cine CT of a beating heart overlayed with velocity field vectors. Levels 5 through 8, frame 1. 75

4.10 Cine CT of a beating heart overlayed with velocity field vectors. Level 4, frames 3 through 8. 76

4.11 Cine CT of a beating heart overlayed with velocity field vectors. Level 5, frames 3 through 8. 77
4.12 Convergence rates of the conjugate gradient algorithm and the preconditioned conjugate gradient algorithm. Solid line—conjugate gradient algorithm. Dotted line—preconditioned conjugate gradient algorithm. 80

5.1 Results of the optimization by dual space method with $(\delta_1, \delta_2) = (.05, 2.5)$ and $(\lambda_1^{(0)}, \lambda_2^{(0)}) = (.02, 60)$. (a) $\lambda_1^{(k)}$ as a function of the iteration number k. (b) $\lambda_2^{(k)}$ as a function of k. (c) The log of the mean squared error in the incompressibility constraint as a function of k. (d) The log of the mean squared error in the divergence-free constraint as a function of k. 94
5.2 Results of the optimization by dual space method with $(\delta_1, \delta_2) = (.05, 2.5)$ and $(\lambda_1^{(0)}, \lambda_2^{(0)}) = (.003, 100)$. (a) $\lambda_1^{(k)}$ as a function of the iteration number k. (b) $\lambda_2^{(k)}$ as a function of k. (c) The log of the mean squared error in the incompressibility constraint as a function of k. (d) The log of the mean squared error in the divergence-free constraint as a function of k. 95
5.3 Results of the optimization by dual space method with $(\delta_1, \delta_2) = (.05, 20)$ and $(\lambda_1^{(0)}, \lambda_2^{(0)}) = (.01, 60)$. (a) $\lambda_1^{(k)}$ as a function of the iteration number k. (b) $\lambda_2^{(k)}$ as a function of k. (c) The log of the mean squared error in the incompressibility constraint as a function of k. (d) The log of the mean squared error in the divergence-free constraint as a function of k. 96
5.4 Results of the optimization by dual space method with $(\delta_1, \delta_2) = (.5, .00001)$ and $(\lambda_1^{(0)}, \lambda_2^{(0)}) = (.01, 60)$. (a) $\lambda_1^{(k)}$ as a function of the iteration number k. (b) $\lambda_2^{(k)}$ as a function of k. (c) The log of the mean squared error in the incompressibility constraint as a function of k. (d) The log of the mean squared error in the divergence-free constraint as a function of k. 97

List of Tables

4.1 Multiplication count for the conjugate gradient algorithm and the preconditioned conjugate gradient algorithm. 78

Abstract

The motion of a deforming body is completely characterized by the velocity field (with initial position) generated by the motion. A method of computing the 3-D velocity field from 3-D *cine CTs* of a beating heart is proposed.

Continuum theory provides two constraints on the velocity field generated by a deforming body. Assuming that *(1)* the image intensity is proportional to some conserved quantity and *(2)* the imaged medium is incompressible, the velocity field must satisfy the *divergence-free constraint* and the *incompressibility constraint.* Computation of the velocity field using these two constraints is an *ill-posed* problem which may be *regularized* using a smoothness term. We define a penalty function as a weighted sum of the two constraining terms and the smoothness term. Minimization of this function yields the desired velocity field. It is shown that, under certain conditions on the image, a unique minimizer of the penalty exists.

Via variational calculus, it can be shown that the solution minimizing the penalty satisfies the Euler-Lagrange equations for this problem. The solution of the Euler-Lagrange equation is a set of coupled elliptic partial differential equations (PDEs). For numerical implementation, the PDEs obtained are discretized resulting in a system of linear equations $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{x}$ is the solution velocity

field. The matrix equation is solved using the conjugate gradient algorithm. Examples of experiments using simulated images and a *cine CT* of a beating heart are presented.

The traditional regularization method does not provide a rigorous approach for obtaining the so-called regularization parameters. For this reason, we reformulate the problem as a constrained minimization. Here, instead of the regularization parameters, we require knowledge of the mean-squared errors of the constraints, which is physically and intuitively more appealing. A solution (and the numerical algorithm) is obtained by the dual space method.

Chapter 1

Introduction

1.1 Motivation and Goals

X-ray computed tomography (CT) is a diagnostic tool for producing cross-sectional images of the human head or body. The reconstructed CT images are proportional to the spatial distribution of the linear x-ray attenuation coefficient within the imaged slice. Since attenuation coefficients vary with tissue type (e.g. blood, muscle, fat, bone), these images yield valuable anatomical information. For most conventional CT scanners the scanning time is on the order of 1 to 10 seconds per slice, consequently, application is limited to parts of the body that are stationary. A modern *cine CT* scanner [1], in which x-rays are produced using an electronically steered electron beam, is able to reduce this scanning time to 50 milliseconds and is therefore ideal for cardiac imaging with minimal motion artifacts. Multiple slices covering the entire heart may be scanned in a few seconds and stored as a sequence of 3-D images of the beating heart. This sequence of images may be displayed on a monitor as a movie for qualitative diagnosis based

on the motion of the heart. The goal of this dissertation is to quantify this motion using the data provided by the cine CT. The motion of a beating heart is completely characterized by the velocity field (with initial position) generated by the motion; it is this field that we propose to estimate.

The recovery of motion from a sequence of images has long been sought [2]. It is often *ill-posed* [3] in the sense of Hadamard [4]. Horn and Schunck [5] reported the first computational algorithm for computing a 2-D velocity field from a sequence of 2-D images using a method commonly referred to as *optical flow*. The ill-posed nature of the problem was overcome via Tikhonov regularization [3]. Several variations on the original optical flow algorithm have since been proposed [6, 7, 8, 9].

The optical flow algorithm of Horn and Schunck [5] computes a velocity vector for every pixel in the image. The *brightness constraint* introduced in [5] is based on the assumption that a 'point' in a sequence of images does not change in its gray level from one frame to the next. This brightness constraint alone however does not result in a unique velocity field. By incorporating a regularization or smoothness measure on the velocity field (thus assuming the true field to be spatially smooth) and minimizing a weighted sum of the smoothness term and the error in the brightness constraint, the 2-D velocity field can be computed from a sequence of 2-D images.

The first application of the optical flow method to recover motion from a sequence of radiographic images appeared in [10]. A modified version of the optical flow algorithm was developed where an extra term—the change in the intensity from one frame to next—was computed and used. The algorithm was shown to be effective for simulated expanding ellipses and for cine angiograms,

both in 2-D.

Mailloux *et al.* [11, 12] have attempted automated motion quantification of a beating heart using echocardiograms. In [11], the optical flow method in [5] was applied directly to 2-D echo images with favorable results. In [12], the velocity field was assumed to be locally linear and the solution constrained to lie on the set of linear vector fields. The linearity constraint, and both the brightness and smoothness constraints of optical flow, can all be shown to be convex. Therefore, by using *projections onto convex sets* (POCS) [13], the velocity field was computed for all components of the linear velocity field: translational, rotational, divergent and shear. One limitation of the results reported by Mailloux *et al.* is that they are 2-D approximations of the true 3-D field.

Since the motion of a beating heart is a 3-D phenomena, we formulate and solve the problem directly in 3-D. The formulation is derived from a physical model for the motion of the imaged medium using continuum theory [14, 15, 16]. Fitzpatrick was the first in suggesting the use of continuum theory for the velocity computation problem given a sequence of images. In [17], he shows that for images proportional to the density of some conserved quantity (e.g. CT images) the *equation of continuity* found in classical continuum theory may be solved to yield the velocity field generated by the motion of the imaged medium. In addition, if the imaged medium is incompressible, the incompressibility condition of continuum theory becomes applicable [18]. The formulation to be presented in the following sections is based on these two conditions.

For a 2-D slice of a 3-D body (e.g. 2-D echocardiograms, or 2-D CT images), the equation of continuity and the incompressibility condition cannot be justified, since in general, and specifically for a beating heart, the motion is not confined

to the 2-D slice being imaged and we would expect the above constraints to be violated. For this reason, motion estimation is addressed here as a 3-D problem.

Specifically, the problem at hand is to determine the time-varying 3-D velocity field given a sequence of 3-D images. The starting point toward a solution is to model the imaged medium as a continuum. In doing so, we have at our disposal, various constitutive equations of the continuum theory.

Continuum theory provides two constraints on the velocity field generated by a deforming body. Assuming that *(1)* the image intensity is proportional to some conserved quantity and *(2)* the imaged medium is incompressible, the velocity field must satisfy the *divergence-free constraint* and the *incompressibility constraint.* Computation of the velocity field using these two constraints is an *ill-posed* problem which may be *regularized* using a smoothness term. We define a penalty function as a weighted sum of the two constraining terms and the smoothness term. Minimization of this function yields the desired velocity field. It is shown that, under certain conditions on the image, a unique minimizer of the penalty exists.

Via variational calculus, it can be shown that the solution minimizing the penalty satisfies the Euler-Lagrange equations for this problem. The solution of the Euler-Lagrange equation is a set of coupled elliptic partial differential equations (PDEs). For numerical implementation, the PDEs obtained are discretized resulting in a system of linear equations $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{x}$ is the solution velocity field. The matrix equation is solved using the conjugate gradient algorithm. Examples of experiments using simulated images and a *cine CT* of a beating heart are presented.

The traditional regularization method does not provide a rigorous approach for

obtaining the so-called regularization parameters. For this reason, we reformulate the problem as a constrained minimization. Here, instead of the regularization parameters, we require knowledge of the mean-squared errors of the constraints, which is physically and intuitively more appealing. A solution (and the numerical algorithm) is obtained by the dual space method.

1.2 Previous Studies on Cardiac Motion

The dynamics of a beating heart is a complicated phenomenon. Methods abound for the analysis of this complex motion. We propose a new method of determining the motion of a beating heart. The proposed method can be used for other images as long as the images are density images of incompressible objects.

Methods for the analysis of cardiac dynamics based on anatomical images may be classified into two basic categories: feature based methods and image intensity based methods. The feature based methods extract points and contours from images and analyze the motion of the features from one frame to the next. The methods that are based on the image intensity do not need to extract features and work with the derivatives of the images.

The motion of a beating heart from radiographic images is difficult to observe due to the lack of visually distinguishable features. Therefore, in [19, 20], radio-opaque markers were placed on the cardiac wall to aid the visualization and to measure the local wall deformation. The markers served as features that can easily be extracted from the images; the extracted features were then used in analyzing cardiac dynamics. Although this is a viable means of understanding the cardiac dynamics, the technique is invasive and is not suitable for human

subjects. Therefore, in [21], natural feature points—bifurcation and inflection points—were selected manually without the use of invasive markers. However, these natural feature points are not in abundance and are difficult to locate.

The methods for analysis that are based on contours (cardiac wall boundary) extract curves and/or surfaces from the images. These methods include solving the 2-D Laplace equation using the extracted 2-D endocardial wall as the boundary (angiogram) [22], construction and analysis of Fourier descriptors for the cardiac surface (ultrasound) [23], and tracking of the ventricular boundary via minimization of the bending energy (MRI) [24]. These techniques ignore the image intensity and only analyze the extracted contour.

The methods based on image intensity compute a velocity vector for every pixel in the image. As mentioned earlier, Cornelius *et al.* [10] and Mailloux *et al.* [11, 12] are such examples. These methods yield 2-D velocity fields for every pixel in the image.

A recent technique, referred to as the MR tagging, places spatial markers or grids non-invasively [25, 26, 27]. The movement of the grid lines on the cardiac wall aids the visualization of the 2-D cardiac wall motion. In [28], a modified version of the optical flow algorithm (an image intensity based method) was used to quantify the motion from 2-D MR tagged images. The drawback of this method is the requirement that T_1 and T_2 are known *a priori*. The knowledge of T_1 and T_2 is necessary to accommodate the decaying magnetic tags during the cardiac cycle.

An emerging technique referred to as phase-contrast MRI [29] measures the velocity directly. This new method, still in the formative stage, can measure the 3-D velocity field with a scan time of four times that of a conventional MR image

[30]. This technique *measures* the 3-D velocity field whereas the proposed method *computes* the 3-D velocity field based on the time-varying gray level image. One advantage of the proposed method is its ability to work with conventional gray level images.

1.3 Overview

This work is organized as follows.

Chapter 2 presents a brief review of continuum theory [14, 15, 16] as it applies to 3-D density images. We develop two constraints on the 3-D velocity field generated by a beating heart. These constraints come from the fact that *1)* the density (CT numbers) is conserved and *2)* the heart muscle do not change in volume during the heart cycle [31].

These two constraints are not sufficient to determine the 3-D velocity field. Therefore, in Chapter 3, the problem of recovering 3-D motion is formulated as an optimization problem where an objective or a cost function is minimized. Then, we pose an unconstrained minimization problem where a minimizer of the penalty function—incorporating the cost and two constraints—is sought as a solution to the motion recovery problem. It is shown that, under certain conditions (usually satisfied by real images), there exists a unique minimizer of this penalty function. The minimizer of the penalty is obtained using the Euler-Lagrange method, and the resulting solution is a set of coupled elliptic PDEs with either the Dirichlet or natural boundary conditions.

Chapter 4 presents a numerical solution. The PDEs are discretized and the effect of this discretization on the existence and uniqueness results of Chapter 3

is discussed. As in the continuous domain, it is shown that there exists a unique discrete solution. Section 4.4 presents the results for both simulated images and clinical cine CT images of a beating heart.

In Chapter 5, we present another approach to the 3-D motion recovery problem. We formulate a constrained optimization problem where the two constraints are used in quadratic form. The dual space method is utilized in developing a solution to this constrained optimization problem. Some results are shown in Section 5.2.3. The chapter concludes with some discussion on the merit of this method.

Chapter 6 contains some concluding remarks and recommendations for further research.

The presentation to follow introduces and defines a large number of variables. To avoid confusion, the variables are listed in Appendix C.

1.4 Contributions

The contributions of this dissertation can be summarized as below.

- Mathematical relationships of continuum theory are reviewed and utilized as constraints on the velocity field generated by the motion of a deforming body.

- The problem of computing 3-D velocity field from sequences of 3-D density images is formulated as an optimization problem minimizing a penalty functional. The unique solution is shown to exist and is obtained from the Euler-Lagrange equations which in this case is a set of coupled elliptic PDEs.

- The numerical algorithm is developed and proved to be useful for simple simulated images and cine CT image of a human heart.

- The divergence-free constraint had been considered and studied in the past [18, 32]. This work is the first successful formulation and implementation incorporating both the incompressibility constraint and the divergence-free constraint for 3-D images.

- Lastly, a method of adaptively updating the regularization (or smoothing) parameters is developed. With numerical examples, it is shown that this method can be made quite effective.

Portions of this dissertation—Chapter 2 in its entirety, parts of Chapter 3 and 4—have appeared in [33, 34].

Chapter 2

Constraints on the Velocity Field of Moving Media

In this chapter we present two constraints which may be used to compute the velocity field from the cine CT of a beating heart. These constraints are developed within the framework of continuum theory. A fundamental assumption in the following is that the data are *density images* in the sense defined by Fitzpatrick [18], i.e., the images represent some conserved quantity. CT image intensities are proportional to the linear attenuation coefficient. This coefficient is a time-invariant function of the tissue type determined by its chemical composition. It is therefore reasonable to assume that the CT image represents a conserved quantity. A mathematical definition of density images and the conservation property is given in chapter 2.2. Although the method described here is derived and applied to x-ray CT images, it can be shown that magnetic resonance (MR) images and emission tomographic images are also density images and hence the algorithm described below could also be applied to images collected using these modalities.

2.1 Descriptions of Motion of Deformable Media

Consider a physical body occupying a region $\mathcal{V} \subset \Re^3$. This body is in motion and is subject to deformation. The region $\mathcal{V}$ consists of points or particles that can be associated with the position vector $\mathbf{R} = (X, Y, Z)$ in one-to-one correspondence. Therefore the mapping 'particle $\rightleftharpoons$ $\mathbf{R}$' is bijective so that each particle is uniquely labeled with a position vector $\mathbf{R}$.

Let a physical body at time t_0 occupy a region $\mathcal{V}_{t_0}$ and at time t, through motion, occupy a new region $\mathcal{V}_t$. Then the particle with label $\mathbf{R} = (X, Y, Z) \in \mathcal{V}_{t_0}$ will have moved to a new position $\mathbf{r} = (x, y, z) \in \mathcal{V}_t$. We describe this mapping by

$$\mathbf{r} = \mathbf{r}(\mathbf{R}, t) \tag{2.1}$$

The mapping $\mathbf{r}(\mathbf{R}, t)$ describes the path of the particle initially located at $\mathbf{R}$. Therefore, it is natural to define the velocity $\mathbf{S}$ at time t of the particle with label $\mathbf{R}$ as follows.

$$\mathbf{S}(\mathbf{R}, t) = \frac{\partial}{\partial t}\mathbf{r}(\mathbf{R}, t) \tag{2.2}$$

Further, we assume that the particle with label $\mathbf{R}$ moves to only one $\mathbf{r}$ and conversely, no two particles with different labels arrive at the same $\mathbf{r}$ at the same time. This assumption is the principle of *impenetrability of matter* [16]. Then, the inverse mapping of (2.1) exists and a pair of invertible mappings are described below.

$$\mathbf{r} = \mathbf{r}(\mathbf{R}, t) \qquad \text{and} \qquad \mathbf{R} = \mathbf{R}(\mathbf{r}, t) \tag{2.3}$$

The above pair of invertible mappings depict the transformation between *spatial description* and *material description.* These are also called *Eulerian* and *Lagrangian* descriptions respectively. In spatial description, the independent variable is $\mathbf{r}$—the spatial variable. In material description, the independent variable is $\mathbf{R}$—the material variable. In both cases, t is an independent variable.

In most imaging experiments, pixels or voxels are fixed to a laboratory frame of reference. The motion of the imaged medium is observed with respect to this laboratory frame in which the pixels are fixed. Therefore, the convenient description of motion in most imaging applications seems to be the spatial description. This is the case for the problem of computing the velocity field within the imaging volume. We must express $\mathbf{S}(\mathbf{R}, t)$ in (2.2) in terms of the spatial variable $\mathbf{r}$—the pixel coordinates.

Using (2.3), the velocity may be expressed in terms of the spatial variable $\mathbf{r}$ as below.

$$\mathbf{s}(\mathbf{r}, t) = \mathbf{S}(\mathbf{R}, t)|_{\mathbf{R}=\mathbf{R}(\mathbf{r},t)} = \mathbf{S}(\mathbf{R}(\mathbf{r}, t), t) \tag{2.4}$$

This is the spatial description of the particle velocity. In other words, $\mathbf{s}(\mathbf{r}, t)$ is the velocity of the particle passing through the spatial position $\mathbf{r}$ at time t.

2.2 The Equation of Continuity

In this section, we present the *continuity equation* using the conservation of mass. Consider a region $\mathcal{V}$ with a density distribution $f(\mathbf{r}, t)$. Let m be the volume integral of f over $\mathcal{V}$. If f represents the mass density then m is the total mass in $\mathcal{V}$. The rate of change in m (within a fixed arbitrary volume $\mathcal{V}$) is given

by

$$\frac{dm}{dt} = -\frac{\partial}{\partial t}\int_{\mathcal{V}} f(\mathbf{r},t)dV \tag{2.5}$$

where

$$dV = \text{differential volume element in } \mathcal{V}$$

This is the change in m as a result of a decrease in density f within $\mathcal{V}$.

Assuming that f is a density of some conserved quantity—meaning that this quantity is neither created nor destroyed—the change in m above should exactly be matched by the flux of m out of the volume $\mathcal{V}$. Mathematically,

$$\frac{dm}{dt} = \oint_{\partial\mathcal{V}} f(\mathbf{r},t)\mathbf{s}(\mathbf{r},t)\cdot d\mathbf{n} \tag{2.6}$$

where

$$\begin{aligned} \partial\mathcal{V} &= \text{surface enclosing } \mathcal{V} \\ d\mathbf{n} &= \text{differential normal surface element on } \partial\mathcal{V} \\ \mathbf{s}(\mathbf{r},t) &= \text{velocity field in spatial description} \end{aligned}$$

Equating (2.5) and (2.6) yields the conservation equation which states that the rate of m (the volume integral of density f) leaving an arbitrary region $\mathcal{V}$ must be canceled by the flux of m across the surface $\partial\mathcal{V}$ enclosing that region

$$\frac{\partial}{\partial t}\int_{\mathcal{V}} f dV + \oint_{\partial\mathcal{V}} f\mathbf{s}\cdot d\mathbf{n} = 0 \tag{2.7}$$

This is the conservation of mass equation (in integral form) that every density image is defined to obey.

Application of the divergence theorem to the flux integral yields

$$\int_{\mathcal{V}} (\frac{\partial}{\partial t} f + \nabla \cdot (f\mathbf{s}))\, dV = 0 \tag{2.8}$$

This must hold for every arbitrary region $\mathcal{V}$. Hence, the integrand itself must be identical to zero.

$$f_t + \nabla \cdot (f\mathbf{s}) = 0 \tag{2.9}$$

This is the conservation of mass equation in differential form. In continuum theory, (2.9) is referred to as the *equation of continuity.* It can be shown that (2.9) holds even for blurred images by defining a blurred version of the velocity field $\mathbf{s}$ [18].

Equation (2.9) may be used as a constraint on the velocity field $\mathbf{s}(\mathbf{r}, t)$. For density images of a compressible medium, the continuity equation (2.9) may be used as a constraint on the velocity field rather than constraints to be discussed in Sections 2.3 and 2.4 to follow. For instance, in [17] and [35], (2.9) was used as a constraint and the problem was solved assuming the velocity field $\mathbf{s}$ to be irrotational. Unfortunately, real velocity fields are rarely curl-free [36].

2.3 The Incompressibility Constraint

The density f may be expressed in either material or spatial descriptions.

$f^L(\mathbf{R}, t)$ in Lagrangian or material description

$f^E(\mathbf{r}, t)$ in Eulerian or spatial description

Recall that in material description, the initial position $\mathbf{R}$ is the independent variable whereas in spatial description, $\mathbf{r}$ is independent. In view of (2.3),

$$f^L(\mathbf{R},t) = f^E(\mathbf{r},t)\Big|_{\mathbf{r}=\mathbf{r}(\mathbf{R},t)}$$

Then, by considering the initial position $\mathbf{R}$ as the fixed variable we take the partial derivative of both sides with respect to t. Using $\mathbf{r} = (x,y,z)$,

$$\begin{aligned}\frac{\partial}{\partial t}f^L(\mathbf{R},t) = \frac{\partial x}{\partial t}\frac{\partial f^E}{\partial x}\Big|_{\mathbf{r}=\mathbf{r}(\mathbf{R},t)} &+ \frac{\partial y}{\partial t}\frac{\partial f^E}{\partial y}\Big|_{\mathbf{r}=\mathbf{r}(\mathbf{R},t)} \\ &+ \frac{\partial z}{\partial t}\frac{\partial f^E}{\partial z}\Big|_{\mathbf{r}=\mathbf{r}(\mathbf{R},t)} + \frac{\partial f^E}{\partial t}\Big|_{\mathbf{r}=\mathbf{r}(\mathbf{R},t)}\end{aligned} \tag{2.10}$$

The partials $\partial x/\partial t, \partial y/\partial t$ and $\partial z/\partial t$ evaluated at $\mathbf{r} = \mathbf{r}(\mathbf{R},t)$ simply represent three components of the velocity in material description $\mathbf{S}(\mathbf{R},t)$. Using (2.3) and (2.4) we may express (2.10) in spatial description as

$$\begin{aligned}&\frac{\partial}{\partial t}f^L(\mathbf{R},t)\Big|_{\mathbf{R}=\mathbf{R}(\mathbf{r},t)} \\ &= \frac{\partial f}{\partial t} + u(\mathbf{r},t)\frac{\partial f}{\partial x} + v(\mathbf{r},t)\frac{\partial f}{\partial y} + w(\mathbf{r},t)\frac{\partial f}{\partial z}\end{aligned} \tag{2.11}$$

where $\mathbf{s}(\mathbf{r},t) = (u,v,w)$. The superscript ^{E} on f have been dropped. From this point on, by f we mean the spatial description of the density—$f^E(\mathbf{r},t)$. The left hand side of (2.11) is the rate of change in the density of the particle initially at $\mathbf{R}$ expressed in terms of the spatial variable $\mathbf{r}$. It is precisely the rate of change of f as seen by an observer moving with the particle initially at $\mathbf{R}$.

The *convected* or *mobile derivative* is defined as the derivative with respect to

time, moving with the particle, as

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \mathbf{s} \cdot \nabla \tag{2.12}$$

Then, we may write (2.11) as

$$\frac{D}{Dt} f = \frac{\partial}{\partial t} f + \mathbf{s} \cdot \nabla f \tag{2.13}$$

For an incompressible medium, the density f does not change in time if the observation is carried out while moving along with the particle. Therefore $D/Dt\ f = 0$ if f represents an incompressible medium. This yields the *incompressibility constraint.*

$$f_t + \nabla f \cdot \mathbf{s} = 0 \tag{2.14}$$

This is equivalent to the *brightness constraint* of optical flow extended to 3-D.

Examples abound where the flow is incompressible and thus satisfying (2.14). For instance, in CT images, the density (and hence CT numbers) of the heart muscle and blood are invariant throughout the systole/diastole cycle. See [31, p. 24] and [37], respectively. Consequently, cine CT images of the human heart should obey the incompressibility constraint (2.14).

2.4 The Divergence-Free Constraint

Thus far, the constraints on the velocity field of a moving body represented by a density image were shown to be the continuity constraint (2.9) and the incompressibility constraint (2.14). These two constraints may be imposed directly;

however, in practice we have found that it was easier to impose the incompressibility constraint and a linear combination of the two constraints.

Equating (2.9) and (2.14) yields,

$$\nabla \cdot (f\mathbf{s}) = \nabla f \cdot \mathbf{s}$$

The left hand side can be expanded as $\nabla \cdot (f\mathbf{s}) = f\,\nabla \cdot \mathbf{s} + \nabla f \cdot \mathbf{s}$, resulting in

$$f\,\nabla \cdot \mathbf{s} = 0 \tag{2.15}$$

In continuum theory, (2.15) is referred to as the *continuity equation for incompressible media.* Equation (2.15) states that for an incompressible medium the divergence of the velocity field must be zero for regions where f is non-zero. That is,

$$\nabla \cdot \mathbf{s} = 0 \tag{2.16}$$

This is the *divergence-free* constraint which incompressible density images must obey. For regions where $f = 0$, where it is void of 'particles', we also assume the velocity field to be divergence-free for mathematical simplicity.

We end this section by noting that when imposing the divergence-free constraint (2.16), the incompressibility constraint (2.14) should also be imposed, since the latter was used in deriving (2.16).

Chapter 3

Motion Recovery from Density Images: Problem Formulation and Solution Methods

3.1 Formulation

We wish to obtain the velocity field $\mathbf{s}$ given a time dependent incompressible density image f. Since the density image f is incompressible, the two constraints discussed in the previous chapter—the incompressibility and the divergence-free constraints—are applicable.

The velocity field $\mathbf{s} = (u, v, w)$ consists of three components. Unfortunately, only two equations (two constraints) are available. It follows that there are infinitely many solutions for this problem. This is a typical ill-posed problem which can be solved via a standard approach commonly referred to as the regularization method.

A brief description of the regularization method is provided in Section 3.1.1. Borrowing the idea of the stabilizing functional, an optimization problem is developed for the motion recovery problem in Section 3.1.2.

3.1.1 Ill-posed Problems and Regularization Methods

Recovery of the velocity field using the incompressibility and the divergence-free constraints is an ill-posed problem, i.e., the solution is not necessarily unique and may be sensitive to small changes in the data. The regularization method developed by Tikhonov [3] overcomes this difficulty by introducing a stabilizing functional for solving such ill-posed problems.

Suppose we wish to solve

$$\mathcal{A}\xi = \eta \tag{3.1}$$

where $\eta \in Z$ is the given data, $\xi \in X$ is the solution vector and $\mathcal{A}$ is an operator from the space X to Z. Both spaces are equipped with a metric.

If equation (3.1) has one and only one solution, i.e., $\mathcal{A}$ is invertible, then the solution is simply

$$\xi = \mathcal{A}^{-1}\eta \tag{3.2}$$

However, if equation (3.1) has no (or many) solution, e.g., $\mathcal{A}$ is singular, then the straight forward method above cannot be used. Moreover, for a computer implementation of the solution to (3.1), we often approximate the operator $\mathcal{A}$ with a finite dimensional matrix A, whether the original operator $\mathcal{A}$ was a matrix or not. Then, even if A is invertible, if it is ill-conditioned, the method (3.2) still breaks down.

Therefore, Tikhonov introduces a stabilizing functional $S(\xi)$ and instead minimizes the following:

$$\mathcal{M} = \mid \mathcal{A}\xi - \eta \mid_Z \quad + \quad \frac{1}{\alpha} S(\xi) \tag{3.3}$$

where $\mid \cdot \mid_Z$ is the metric in the space Z, and α is the so-called regularization parameter. Then, the solution ξ_0 is the minimizer of the functional above. The regularization parameter α can be obtained if one knows the discrepancy $\mid \mathcal{A}\xi - \eta \mid_Z$.

The regularizing operator $\mathcal{R}(\alpha)$ is then defined as mapping from the space Z to X as below:

$$\xi_0 = \mathcal{R}(\alpha)\eta$$

Note that the regularizing operator $\mathcal{R}(\alpha)$ depends on the choice of the regularization parameter α. Therefore, the solution may be different for different values of α. The choice of α is an important theoretical question and will be discussed in Chapter 5 where we deal with two such parameters.

Assuming the space X to be some function space such as the Hilbert space $L_2([a,b])$ or the space of continuous functions $C([a,b])$, Tikhonov suggests the following stabilizing functional:

$$S(\xi) = \int_a^b \sum_{r=0}^{p} q_r(x) \left(\frac{d^r \xi}{dx^r} \right)^2 dx \; . \tag{3.4}$$

In the computer vision literature, the most commonly used stabilizer is of the form (3.4) with $p = 1, q_0 = 0$ and $q_1 = 1$ extended to 2-D [38]:

$$S(\xi_1, \xi_2) = \int_a^b \int_c^d \left(\left(\frac{\partial \xi_1}{\partial x} \right)^2 + \left(\frac{\partial \xi_1}{\partial y} \right)^2 + \left(\frac{\partial \xi_2}{\partial x} \right)^2 + \left(\frac{\partial \xi_2}{\partial y} \right)^2 \right) dx \, dy \; . \tag{3.5}$$

where ξ_1 and ξ_2 are some scalar functions on spaces such as $C([a,b] \times [c,d])$. In optical flow, this stabilizer is referred to as the *smoothness constraint.* In the following section, the optimization problem will be developed using the stabilizing functional (3.4) as the cost functional.

3.1.2 Optimization Approach

In light of the previous section and its stabilizing functional, we introduce the following stabilizer

$$e_S(\mathbf{s}) = \int_\Omega (u_x^2 + u_y^2 + u_z^2 + v_x^2 + v_y^2 + v_z^2 + w_x^2 + w_y^2 + w_z^2)\, d\Omega \qquad (3.6)$$

where

$$\begin{aligned} \Omega &= \{(x,y,z) \in \Re^3 \ : \ 0 \le x \le T_x,\ 0 \le y \le T_y,\ 0 \le z \le T_z\} \\ d\Omega &= dx\, dy\, dz \\ \mathbf{s} &= (u,v,w) \\ \Re &= \text{the real line} \end{aligned}$$

and (T_x, T_y, T_z) is the spatial extent of the imaging volume.

Combining the functional e_S (3.6) and the two constraints, we pose the following optimization problem:

$$\begin{aligned} &\textit{minimize} && e_S(\mathbf{s}) \\ &\textit{subject to} && f_x u + f_y v + f_z w + f_t = 0 \quad \text{and} \quad u_x + v_y + w_z = 0 \end{aligned} \qquad (3.7)$$

Rather than solving the above optimization problem directly, we consider the

equivalent problem.

$$\begin{aligned} &\textit{minimize} && e_S(\mathbf{s}) \\ &\textit{subject to} && e_I(\mathbf{s}) = 0 \quad \text{and} \quad e_D(\mathbf{s}) = 0 \end{aligned} \tag{3.8}$$

where

$$\begin{aligned} e_I(\mathbf{s}) &= \int_\Omega (f_x u + f_y v + f_z w + f_t)^2 \, d\Omega \\ e_D(\mathbf{s}) &= \int_\Omega (u_x + v_y + w_z)^2 \, d\Omega \end{aligned}$$

Since (3.8) is a convex minimization problem over convex constraints, one can apply the method of Lagrange multipliers to find an optimal solution. In Chapter 5, we will revisit this problem of finding the Lagrange multipliers via the dual-space method.

In this section, we confine our attention to finding an approximate solution to (3.8) by unconstrained minimization of the penalty function

$$e(\mathbf{s}) = e_S(\mathbf{s}) + \gamma_1 \, e_I(\mathbf{s}) + \gamma_2 \, e_D(\mathbf{s}) \tag{3.9}$$

where γ_1 and γ_2 are a pair of real positive constants. By way of justification of this approximate solution, we note that in practice, the data contains noise, and the use of hard constraints as in (3.7) may result in poor solutions due to the incorporation of the noise into the computed velocity fields. In contrast, the solution obtained by minimizing (3.9) does not require that the constraints be exactly met, and consequently may be more robust to noise.

In the optical flow formulation of Horn and Schunck [5], a global constraint—a

2-D version of e_S above—was introduced so that a solution may be obtained. The penalty method presented in this section is identical to their approach. In fact, if $\gamma_2 = 0$, our method yields a straight-forward extension of optical flow solution [5] to 3-D.

Before proceeding to solve the optimization problem of minimizing $e(\mathbf{s})$ of (3.9), it desirable to know whether $e(\mathbf{s})$ is convex or not. If $e(\mathbf{s})$ is a convex functional, then any local minimum turns out to be a global minimum. Indeed, $e(\mathbf{s})$ is convex; and strictly convex under a certain condition on f. The mathematics pertaining to the proof of this assertion is the topic of next two sections.

3.2 Convexity of Functionals on a Vector Space

This section presents some fundamental definitions and theorems of functional analysis that are relevant to our discussion. In the presentation, only the results necessary to show the convexity of $e(\mathbf{s})$ are provided. Accordingly, readers are referred to [39] or [40] for a through treatment of this material.

Definition 3.1 Let X be a vector space, g a functional on subset Ψ of X, and α a scalar. For a fixed $\xi \in \Psi$ and an arbitrary $h \in X$, if the limit

$$\delta g(\xi; h) = \lim_{\alpha \to 0} \frac{g(\xi + \alpha h) - g(\xi)}{\alpha} \tag{3.10}$$

exists, it is called the *Gateaux differential of g at ξ with increment h*. Moreover, if the limit above exists for every $h \in X$, then g is said to be *Gateaux differentiable at ξ*.

Note that if g is Gateaux differentiable at ξ, then (3.10) can be written as an ordinary derivative

$$\delta g(\xi; h) = \frac{d}{d\alpha} g(\xi + \alpha h)\Big|_{\alpha=0} \tag{3.11}$$

Equation (3.11) becomes useful in the proofs of Lemma 3.6 and Theorem 3.7 as well as in deriving the Euler-Lagrange equation for optimizing the penalty function $e(\mathbf{s})$ (see Section 3.3 and Appendix A.1). We show a usage of (3.11) in the following example.

Example 3.2 Consider the functional $J : \mathbf{C}(\Omega) \mapsto \Re$

$$J(\mathbf{s}) = \int_\Omega g(\mathbf{s}) d\Omega$$

where

$$\begin{aligned} C(\Omega) &= \{\text{continuous functions on } \Omega\} \\ \mathbf{C}(\Omega) &= C(\Omega) \times C(\Omega) \times C(\Omega) \\ \mathbf{s} &= (u, v, w) \in \mathbf{C}(\Omega) \end{aligned}$$

and $\Omega \subset \Re^3$ as before. Then, by (3.11), the Gateaux differential

$$\delta J(\mathbf{s}; \mathbf{h}) = \frac{d}{d\alpha} \int_\Omega g(\mathbf{s} + \alpha \mathbf{h}) d\Omega \Big|_{\alpha=0}.$$

By assuming that the partial derivatives, g_u, g_v and g_w exist, the interchange of differentiation and integration is permissible. Hence, by using

$$\delta g(\mathbf{s}; \mathbf{h}) = g_u h_1 + g_v h_2 + g_w h_3$$

it follows that

$$\delta J(\mathbf{s};\mathbf{h}) = \int_{\Omega} \{g_u h_1 + g_v h_2 + g_w h_3\} d\Omega \tag{3.12}$$

where $\mathbf{s} = (u, v, w), \mathbf{h} = (h_1, h_2, h_3)$.

In Appendix A.1, g is slightly more complicated but the approach in obtaining the Gateaux differential is identical.

For the discussion on existence and the uniqueness of the solution, the concept of convexity becomes essential. We state a standard definition on convexity.

Definition 3.3 A functional g defined on a convex subset C of a vector space X is said to be *convex on* C, if for $\beta \in (0, 1)$,

$$g(\beta\xi + (1-\beta)\eta) \leq \beta g(\xi) + (1-\beta) g(\eta) \qquad \forall \xi, \eta \in C \tag{3.13}$$

If the inequality is strict, for every $\xi \neq \eta$, then g is said to be *strictly convex.*

We state an immediate consequence of Definition 3.3 as a lemma. The proof is trivial and thus omitted.

Lemma 3.4 *Let g_1 and g_2 be convex functionals on a convex subset C of vector space X. Then, the sum $g_1 + g_2$ is convex on C. Moreover, if either g_1 or g_2 is strictly convex on C, then the sum $g_1 + g_2$ is strictly convex on C.*

The Definition 3.3 is not a useful one when applying to functionals of interest—e_S, e_I, e_D and e. Therefore, we state an equivalent definition, which is applicable to these functionals.

Lemma 3.5 *A Gateaux differentiable functional g on a convex subset C of a vector space X is convex on C iff*

$$g(\eta) \geq g(\xi) + \delta g(\xi; \eta - \xi) \qquad \forall \xi, \eta \in X \tag{3.14}$$

where $\delta g(\xi; \eta - \xi)$ is the Gateaux differential at ξ with increment $(\eta - \xi)$. If the inequality is strict, for every $\xi \neq \eta$, then g is strictly convex on X.

The proof is provided in the Appendix B. Also see [41].

With this result, it will be shown that the functional e is convex, and strictly convex under certain conditions on f. With strict convexity of e, the existence and uniqueness of the minimum follows.

3.3 Existence and Uniqueness of the Solution

Before the discussion of the properties of various functionals of interest, the specification of the domain of these functionals is needed. The choice of the space is arbitrary; however, it is necessary to fit into the framework of the functional analysis. We define the following normed spaces.

$$\begin{aligned} \mathbf{C}(\Omega) &= C(\Omega) \times C(\Omega) \times C(\Omega) \\ \mathbf{C}^1(\Omega) &= C^1(\Omega) \times C^1(\Omega) \times C^1(\Omega) \end{aligned}$$

where [40]

$$\begin{aligned} C(\Omega) &= \{\text{continuous functions on } \Omega\} \\ C^1(\Omega) &= \{\text{continuously differentiable functions on } \Omega\} \end{aligned}$$

The functionals e_S, e_I and e_D may now be formally defined as mappings as below.

$$e_S : \mathbf{C}^1(\Omega) \longrightarrow \Re \tag{3.15}$$

$$e_S : (u,v,w) \longmapsto \int_\Omega (u_x^2 + u_y^2 + u_z^2 + v_x^2 + v_y^2 + v_z^2 + w_x^2 + w_y^2 + w_z^2)\, d\Omega$$

$$e_I : \mathbf{C}(\Omega) \longrightarrow \Re \tag{3.16}$$

$$e_I : (u,v,w) \longmapsto \int_\Omega (f_x u + f_y v + f_z w + f_t)^2\, d\Omega$$

$$e_D : \mathbf{C}^1(\Omega) \longrightarrow \Re \tag{3.17}$$

$$e_D : (u,v,w) \longmapsto \int_\Omega (u_x + v_y + w_z)^2\, d\Omega$$

The above functionals are all quadratic and convexity follows. We state this assertion as a lemma below.

Lemma 3.6 *The functionals e_S, e_I and e_D as defined in* (3.15), (3.16) *and* (3.17) *are all convex on their respective domains*

The proof is provided in the Appendix B.

Since $e(\mathbf{s})$ is a sum of convex functionals, by Lemma 3.4, e is convex itself. In addition, e is strictly convex if the components of the gradient of the given image f are linearly independent for each t. This assertion is proved in Theorem 3.7.

Theorem 3.7 *The functional $e = e_S + \gamma_1 e_I + \gamma_2 e_D, \quad \gamma_1, \gamma_2 \geq 0$ is convex. Moreover, if the given image $f(x,y,z,t)$ is such that the components of $\nabla f = (f_x, f_y, f_z)$ are linearly independent on $\mathbf{V} = [0,T_x] \times [0,T_y] \times [0,T_z]$ for each fixed t, then e is strictly convex.*

Proof:

Since a sum of convex functionals yields another convex functional, e is convex

as e_S, $\gamma_1 e_I$ and $\gamma_2 e_D$ are all convex by Lemma 3.6. It suffices to show that e is strictly convex if the components of ∇f are linearly independent.

Let $J(\mathbf{s}) = e_S(\mathbf{s}) + \gamma_1 e_I(\mathbf{s})$. Showing that J is strictly convex, completes the proof by Lemma 3.4, as e is a sum of J and e_D. Therefore, we establish

$$J(\mathbf{s}_2) > J(\mathbf{s}_1) + \delta J(\mathbf{s}_1; \mathbf{s}_2 - \mathbf{s}_1) \tag{3.18}$$

assuming linear independence of the components of ∇f.

From the definition of Gateaux differential (see Appendix B),

$$\begin{aligned} \delta e_S(\mathbf{s};\mathbf{h}) &= 2\int_\Omega \{(u_x h_{1x} + u_y h_{1y} + u_z h_{1z}) + (v_x h_{2x} + v_y h_{2y} + v_z h_{2z}) \\ &\qquad + (w_x h_{3x} + w_y h_{3y} + w_z h_{3z})\}\, d\Omega \qquad (3.19) \\ \delta e_I(\mathbf{s};\mathbf{h}) &= 2\int_\Omega \{(f_x u + f_y v + f_z w)(f_x h_1 + f_y h_2 + f_z h_3)\}\, d\Omega \end{aligned}$$

Therefore, in view of (B.4) and (B.8), equation (3.18) can be shown to be equivalent to

$$\begin{aligned} 0 < \int_\Omega \{&(u_{2x} - u_{1x})^2 + (u_{2y} - u_{1y})^2 + (u_{2z} - u_{1z})^2 \qquad (3.20) \\ &+ (v_{2x} - v_{1x})^2 + (v_{2y} - v_{1y})^2 + (v_{2z} - v_{1z})^2 \\ &+ (w_{2x} - w_{1x})^2 + (w_{2y} - w_{1y})^2 + (w_{2z} - w_{1z})^2\}\, d\Omega \\ &+ \gamma_1 \int_\Omega \{f_x(u_2 - u_1) + f_y(v_2 - v_1) + f_z(w_2 - w_1)\}^2\, d\Omega \end{aligned}$$

For notational convenience, let

$$g(\mathbf{s}) = \int_\Omega (f_x u + f_y v + f_z w)^2\, d\Omega$$

The null spaces of e_S and g can be shown to be

$$\begin{aligned} \mathcal{N}(e_S) &= \{\mathbf{s} = (u, v, w): \quad u, v, w \quad \text{are constants in } (x, y, z)\} \\ \mathcal{N}(g) &= \{\mathbf{s} = (u, v, w): \quad f_x u + f_y v + f_z w = 0\} \end{aligned} \tag{3.21}$$

The right hand side (RHS) of (3.20) is at least zero or bigger since all terms are quadratic. Therefore, showing that RHS $\neq 0$ if the components of ∇f are linearly independent completes the proof. We show the contrapositive.

Assume that RHS is zero. Then, for a fixed t there exists an element

$$\mathbf{c} = (c_1, c_2, c_3) \in \mathbf{C}^1(\Omega)$$

where

$$(c_1(x, y, z, t), c_2(x, y, z, t), c_3(x, y, z, t)) \in \Re^3$$

such that

$$\mathbf{c} \in \mathcal{N}(e_S) \cap \mathcal{N}(g) - \{(0, 0, 0)\}.$$

Then, from $\mathbf{c} \in \mathcal{N}(e_S)$ we require c_1, c_2, c_3, not all zero vectors, to be constants on Ω. Moreover, $\mathbf{c} \in \mathcal{N}(g)$ implies $f_x c_1 + f_y c_2 + f_z c_3 = 0$. Therefore, we have:

$$\nabla f \cdot \mathbf{c} = 0 \qquad \text{where } \mathbf{c} \neq (0, 0, 0) \text{ is some constant function on } \Omega \tag{3.22}$$

In other words, the components of ∇f are linearly dependent, which establishes the contrapositive. Therefore, unless the components of the gradient of f are linearly dependent, equation (3.20) holds and thus J is strictly convex. This completes the proof. See [42] for a 2-D version of this theorem and the proof

without the use of Gateaux differentials.

Physically, if the components of ∇f are linearly dependent, then f is an image whose contours of constant gray levels are parallel lines if f is a 2-D image. For 3-D images, linearly dependent ∇f confines ∇f to a plane. These images are not realistic images and in all cases, except for carefully constructed images, the components of ∇f are usually linearly independent.

It is well known that for finite dimensional problems, convexity of a functional guarantees a solution and strict convexity of a functional guarantees uniqueness in addition to existence. This can be generalized to infinite dimensional vector spaces as shown in the Lemma [41] below.

Lemma 3.8 *If a functional g on a bounded convex subset C of a vector space X is lower semi-continuous, then 1) If g is convex, there exists a minima x_0 of g. 2) If g is strictly convex, the minima x_0 of g is unique.*

Lemma 3.8 guarantees uniqueness and existence of a solution $\mathbf{s_0}$ minimizing the penalty $e(\mathbf{s})$.

We note however, the penalty functional $e(\mathbf{s})$ depends on the parameters γ_1 and γ_2. Therefore, the 'unique solution' does exist but it depends on these parameters. A method of computing these parameters uniquely will be addressed in Chapter 5.

3.4 Solution by Euler-Lagrange Method

In this section, we present a solution minimizing the penalty functional $e(\mathbf{s})$.

$$e(\mathbf{s}) = \int_\Omega \{(u_x^2 + u_y^2 + u_z^2 + v_x^2 + v_y^2 + v_z^2 + w_x^2 + w_y^2 + w_z^2) \qquad (3.23)$$
$$+ \ \gamma_1 \left(f_x u + f_y v + f_z w + f_t\right)^2 \ + \ \gamma_2 \left(u_x + v_y + w_z\right)^2\} \, d\Omega$$

where $\mathbf{s} = (u, v, w)$ and γ_1 and γ_2 are real positive constants.

Let F be the integrand of (3.23). Then, from the calculus of variations [43] the solution $\mathbf{s}_0$ must satisfy the following set of Euler-Lagrange equations with either the Dirichlet or the natural boundary conditions (see Appendix A.1).

$$\begin{aligned} 0 &= F_u - \frac{\partial}{\partial x}F_{u_x} - \frac{\partial}{\partial y}F_{u_y} - \frac{\partial}{\partial z}F_{u_z} \\ 0 &= F_v - \frac{\partial}{\partial x}F_{v_x} - \frac{\partial}{\partial y}F_{v_y} - \frac{\partial}{\partial z}F_{v_z} \\ 0 &= F_w - \frac{\partial}{\partial x}F_{w_x} - \frac{\partial}{\partial y}F_{w_y} - \frac{\partial}{\partial z}F_{w_z} \end{aligned} \tag{3.24}$$

From the definition of F,

$$\begin{aligned} F_u &= 2\,\gamma_1 f_x(f_x u + f_y v + f_z w + f_t) \\ \frac{\partial}{\partial x}F_{u_x} &= 2\,u_{xx} - 2\,\gamma_2\,(u_{xx} + v_{xy} + w_{xz}) \\ \frac{\partial}{\partial y}F_{u_y} &= 2\,u_{yy} \\ \frac{\partial}{\partial z}F_{u_z} &= 2\,u_{zz} \end{aligned}$$

The substitution of the partials into the Euler-Lagrange equation (3.24) results in the following set of partial differential equations (PDEs).

$$\begin{aligned} \nabla^2 u &= \gamma_1\, f_x(f_x u + f_y v + f_z w + f_t) \;-\; \gamma_2\,(u_{xx} + v_{xy} + w_{xz}) \\ \nabla^2 v &= \gamma_1\, f_y(f_x u + f_y v + f_z w + f_t) \;-\; \gamma_2\,(u_{xy} + v_{yy} + w_{yz}) \\ \nabla^2 w &= \gamma_1\, f_z(f_x u + f_y v + f_z w + f_t) \;-\; \gamma_2\,(u_{xz} + v_{yz} + w_{zz}) \end{aligned} \tag{3.25}$$

where ∇^2 is the Laplacian operator. The solution satisfies equation (3.25) on the interior of Ω. On the boundary $\partial\Omega$, we impose either the Dirichlet or natural

boundary conditions (see Appendix A.1).

If we let $\gamma_2 = 0$, in (3.25) then the problem is identical to the optical flow problem [5] extended to 3-D; and the solution satisfies the PDEs:

$$\begin{aligned} \nabla^2 u &= \gamma_1 f_x(f_x u + f_y v + f_z w + f_t) \\ \nabla^2 v &= \gamma_1 f_y(f_x u + f_y v + f_z w + f_t) \\ \nabla^2 w &= \gamma_1 f_z(f_x u + f_y v + f_z w + f_t) \end{aligned} \tag{3.26}$$

As in [5] the Laplacian may be discretized as $\nabla^2 g = \kappa(\bar{g} - g)$ where κ is a constant depending on the differential mask and $\bar{g}$ is the local average of g. Algebraic manipulation and a symbolic inversion of the 3×3 matrix result in a Jacobi type iterative algorithm.

$$\begin{aligned} u^{(n+1)} &= \bar{u}^{(n)} - f_x \frac{\bar{u}^{(n)} f_x + \bar{v}^{(n)} f_y + \bar{w}^{(n)} f_z + f_t}{(\kappa/\gamma_1) + f_x^2 + f_y^2 + f_z^2} \\ v^{(n+1)} &= \bar{v}^{(n)} - f_y \frac{\bar{u}^{(n)} f_x + \bar{v}^{(n)} f_y + \bar{w}^{(n)} f_z + f_t}{(\kappa/\gamma_1) + f_x^2 + f_y^2 + f_z^2} \\ w^{(n+1)} &= \bar{w}^{(n)} - f_z \frac{\bar{u}^{(n)} f_x + \bar{v}^{(n)} f_y + \bar{w}^{(n)} f_z + f_t}{(\kappa/\gamma_1) + f_x^2 + f_y^2 + f_z^2} \end{aligned} \tag{3.27}$$

If the most recent updated values are used in the iteration above, we obtain a Gauss-Seidel iteration. The successive over-relaxation (SOR) method [44] may also be used.

In Jacobi type iterations, convergence is guaranteed if the *row-sum criterion* [44] is met. Unfortunately, the row-sum criterion cannot be checked since the row elements depend on the image f. However, we have implemented (3.27) in 3-D and obtained convergence for a large class of images. It is more difficult to obtain

a Jacobi type iterative formula for the case $\gamma_2 \neq 0$ as this involves a symbolic inversion of a more complex 3×3 matrix. Although this symbolic inversion can be done, we chose to use the conjugate gradient algorithm where convergence is guaranteed [45, 46].

3.5 Boundary Conditions

The solution vector (u, v, w) cannot be obtained from (3.25) without a boundary condition. We present two standard boundary conditions below. These boundary conditions in relation to the Euler-Lagrange equation (3.24) is described in Appendix A.1.

3.5.1 Dirichlet Boundary Condition

In certain applications, the values of (u, v, w) on $\partial\Omega$ may be known *a priori.* This is called the *essential* or the *Dirichlet boundary condition.* Then, the problem is completely specified and the solution (u, v, w) may be obtained on the interior of Ω.

However, we must revisit the uniqueness and the existence results of Section 3.3. Theorem 3.7 mentions no boundary conditions and is proved without any boundary conditions. By placing an essential boundary condition such as

$$(u, v, w) = (b_1, b_2, b_3) \text{ on } \partial\Omega$$

we are placing a restriction on the solution space $\mathbf{C}^1(\Omega)$. Therefore, the proof of Theorem 3.7 must be revisited. We note that since the solution is assumed to exist

in the space $\mathbf{C}^1(\Omega)$, the functions (b_1, b_2, b_3) must be continuously differentiable on $\partial\Omega$.

Recall that Theorem 3.7 was proved by noting that if the the components of ∇f are linearly independent, the space

$$\mathcal{N} = \{\mathbf{c} \in \mathbf{C}^1(\Omega) : \mathbf{c} \in \mathcal{N}(e_S) \cap \mathcal{N}(g) - \{(\mathbf{0}, \mathbf{0}, \mathbf{0})\}\}$$

is empty which enforces the RHS of equation (3.20) to be strictly positive. See equation (3.21) for definitions of $\mathcal{N}(e_S)$ and $\mathcal{N}(g)$.

Here, we are placing a restriction on the solution space, via the essential boundary condition; and there are less elements in the space $\mathcal{N}$ that can force the RHS of equation (3.20) to be zero. Therefore the RHS of equation (3.20) is guaranteed to be strictly positive if the components of ∇f are linearly independent. Hence, with the Dirichlet boundary condition, there exists a unique minimizer of the penalty functional e, if the components of ∇f are linearly independent.

Depending on the functions (b_1, b_2, b_3) the linear independence condition may not be needed. For instance, if any one of the functions, b_1, b_2 or b_3, is not a constant function on $\partial\Omega$, then the space $\mathcal{N}$ is empty without any condition on ∇f. This forces the RHS of equation (3.20) to be strictly positive and the strict convexity follows without any condition on f.

We conclude this section by noting that the optimum solution with the Dirichlet boundary condition is not the optimum solution shown to exist by Theorem 3.7. The solution shown to exist in Theorem 3.7 is the solution with no knowledge of the values of (u, v, w) on $\partial\Omega$ (natural boundary condition). In general, the optimum solution with the Dirichlet boundary condition will attain a larger penalty

e than the optimum solution with the natural boundary condition. The solution with the natural boundary condition will be discussed in the next section.

3.5.2 Natural Boundary Condition

If we have no knowledge of the values of (u, v, w) on $\partial\Omega$, this condition is called the *natural boundary condition.* Unlike the Dirichlet boundary condition case, we note that Theorem 3.7 is fully applicable here without any modifications.

Using the natural boundary conditions (A.3), we have (see Appendix A.1 for a detailed derivation):

$$\left.\begin{aligned} u_x + \gamma_2(u_x + v_y + w_z) &= 0 \\ v_x &= 0 \\ w_x &= 0 \end{aligned}\right\} \quad \text{at } x = 0, \text{ and } x = T_x$$

$$\left.\begin{aligned} u_y &= 0 \\ v_y + \gamma_2(u_x + v_y + w_z) &= 0 \\ w_y &= 0 \end{aligned}\right\} \quad \text{at } y = 0, \text{ and } y = T_y \qquad (3.28)$$

$$\left.\begin{aligned} u_z &= 0 \\ v_z &= 0 \\ w_z + \gamma_2(u_x + v_y + w_z) &= 0 \end{aligned}\right\} \quad \text{at } z = 0, \text{ and } z = T_z$$

The natural boundary condition is not well-defined at edges or vertices of Ω, as the normal vector $\mathbf{n}$ is not continuous there. At an edge we may insist that boundary conditions corresponding to the two intersecting boundary planes both be satisfied; and at a corner, we impose three boundary conditions for the three intersecting boundary planes.

The above natural boundary conditions together with the PDEs (3.25) completely specify the system to be solved to find the desired velocity field $\mathbf{s} = (u, v, w)$.

3.6 Discussion

In this chapter, we have studied the problem of 3-D motion recovery given a sequence of 3-D density images. The penalty functional was developed using a standard optimization technique using the idea of stabilizing functionals found in regularization theory. The penalty functional was shown to be convex and strictly convex if the components of ∇f are linearly independent. The existence and the uniqueness of the solution followed. The linearly independent condition is not a severe one as all realistic images should obey this linearly independence condition.

The solution was obtained for the minimization problem using the Euler-Lagrange equations for this problem. The Dirichlet boundary condition and its significance on the existence and the uniqueness of the solution was discussed. The natural boundary condition was also provided for the case where nothing about the solution is known on the boundary.

There are no closed form solutions for the resulting PDEs (3.25) with their associated boundary conditions (Dirichlet or natural). A numerical solution must be sought. In Chapter 4, we discretize the PDEs (3.25) and present the conjugate gradient algorithm to solve the resulting linear system of equations.

The choice of the parameters γ_1 and γ_2 still remains an important question. These were treated as predetermined positive constants in this chapter. Determining these parameters (sometimes called regularization parameters) remains an

active research area and in Chapter 5 we present a method to adaptively update these parameters based on the mean-squared errors in the constraints.

Chapter 4

Solution in the Discrete Domain: Numerical Methods

In Chapter 3, we have presented a formulation and its solution for the problem of 3-D motion recovery from 3-D density images. The resulting solution was a set of coupled elliptic PDEs which has no known analytical solution. The numerical solution to the non-coupled elliptic PDEs is well-developed [47] and the solution to (3.25) presented in this chapter follows closely to that approach. The chapter begins with the discretization of the PDEs (3.25) and its dependence on the boundary condition imposed.

It is important that the devised algorithm converges to a unique solution. Otherwise, we have no confidence in the solution that the algorithm provides. For this reason, in Section 4.2, we discuss the existence and the uniqueness of the discrete solution to (3.25). Although we have provided the existence and the uniqueness proof for the problem in the continuous domain in Chapter 3, the discretization process sometimes alters the criteria in which a unique solution

exists. We address this question in Section 4.2 in the discrete domain.

Section 4.3 presents the conjugate gradient algorithm which solves (3.25) numerically. The preconditioned conjugate gradient algorithm is also discussed here. Then, in Section 4.4, we present some results obtained using the conjugate gradient algorithm for several simulated images and a cine CT of a beating heart.

4.1 Discretization of the PDEs

To compute a solution for the PDEs in (3.25), the equations must be discretized. Assuming uniform sampling, let the spatial sample grid spacings be Δ_x, Δ_y, and Δ_z for the x, y and z-axis respectively, and let

$$f_{ijk} = f(x,y,z)|_{(x,y,z)=(i\,\Delta_x,\, j\,\Delta_y,\, k\,\Delta_z)}, \qquad (x,y,z) \in \Omega \tag{4.1}$$

The partial derivatives (f_x, f_y, f_z, f_t) and the velocity components (u, v, w) are similarly discretized.

Using lexicographical ordering [48], the image samples f_{ijk} can be vector-stacked, e.g., $\mathbf{f} = [\ \cdots \quad f_{ijk} \quad \cdots\ ]^T$. The vectors, $\mathbf{f}_x, \mathbf{f}_y, \mathbf{f}_z, \mathbf{f}_t$, $\mathbf{u}$, $\mathbf{v}$ and $\mathbf{w}$ are similarly constructed. The solution vector $\mathbf{x}$ is then defined as

$$\mathbf{x} = \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \\ \mathbf{w} \end{bmatrix} \tag{4.2}$$

To express the PDEs of (3.25) in the discrete domain, the matrices below are

defined.

$$
\begin{aligned}
\mathbf{H}_D &= [\,\mathbf{D}_x \mid \mathbf{D}_y \mid \mathbf{D}_z\,] \\
\mathbf{H}_I &= [\,\mathrm{diag}(\mathbf{f}_x) \mid \mathrm{diag}(\mathbf{f}_y) \mid \mathrm{diag}(\mathbf{f}_z)\,] \\
\mathbf{Q} &= \begin{bmatrix} \mathcal{D} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathcal{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{D} \end{bmatrix} \\
\mathcal{D} &= \text{the (negative) Laplacian matrix} \\
\mathrm{diag}(\mathbf{f}_x) &= \text{a diagonal matrix with elements of } \mathbf{f}_x \text{ in the diagonal}
\end{aligned}
\tag{4.3}
$$

where $\mathbf{D}_x, \mathbf{D}_y$ and $\mathbf{D}_z$ are matrix representations of partial differential operators with respect to x, y and z respectively. With this discretization, (3.25) has the following discrete form.

$$
\mathbf{A}\mathbf{x} = \mathbf{b}, \qquad \text{where} \quad \begin{cases} \mathbf{A} &= \mathbf{Q} + \gamma_1\, \mathbf{H}_I^T\, \mathbf{H}_I + \gamma_2\, \mathbf{H}_D^T\, \mathbf{H}_D \\ \mathbf{b} &= -\gamma_1\, \mathbf{H}_I^T\, \mathbf{f}_t \end{cases} \tag{4.4}
$$

At this point, equation (4.4) and all matrices therein, are not completely specified. The exact form of the matrices will be determined by the type of boundary condition that we enforce. It also depends on the choice of the differentiation mask. Even the dimension of equation (4.4) depends on the choice of the boundary condition.

Since the choice of the boundary condition and the discretization mask plays an important role in determining the structure of $\mathbf{A}$, and also to establish the notation for Section 4.2, we address the effects of different boundary conditions on equation (4.4).

4.1.1 Dirichlet Boundary Condition

If we know the value of $\mathbf{s}$ on the boundary $\partial\Omega$, then the natural choice is the Dirichlet boundary condition. For instance, the motion $\mathbf{s}$ may be known *a priori* on the bounding planes of the 3-D image. In this case, the value of $\mathbf{x}$, the discrete version of $\mathbf{s}$, is known on $\partial\Omega$. Hence, the matrix equation (4.4) is of dimension $3N_d \times 3N_d$ where $N_d = (N_x - 2) \times (N_y - 2) \times (N_z - 2)$, where we have discarded all the elements corresponding to the boundary $\partial\Omega$. The algorithm only computes elements of $\mathbf{x}$ corresponding to the interior of Ω. Therefore, the differential operators of (4.4)—assuming the usual central difference derivative, seven point Laplacian, etc—can be defined everywhere in the domain of interest and (4.4) may be solved.

We will represent this new equation as below:

$$\mathbf{A}_d\mathbf{x}_d = \mathbf{b}_d, \qquad \text{where} \quad \begin{cases} \mathbf{A}_d &= \mathbf{Q}_d + \gamma_1\, \mathbf{H}_{Id}^T\, \mathbf{H}_{Id} + \gamma_2\, \mathbf{H}_{Dd}^T\, \mathbf{H}_{Dd} \\ \mathbf{b}_d &= -\gamma_1\, \mathbf{H}_{Id}^T\, \mathbf{f}_{td} \end{cases} \tag{4.5}$$

where the subscript $_d$ indicate a modified version of the appropriate matrices or vectors of (4.4) to reflect the Dirichlet boundary condition. We note that $\mathbf{A}_d$ is symmetric and we will show that it is actually positive definite in Section 4.2.1.

4.1.2 Neumann Boundary Condition

If we have no knowledge on the value of $\mathbf{s}$ on the boundary $\partial\Omega$, we must resort to the natural boundary condition. The natural boundary condition (3.28) of Section 3.5.2 is difficult to implement and the resulting system matrix $\mathbf{A}$ is difficult to analyze.

We modify the natural boundary condition slightly, and impose the condition:

$$\nabla \cdot \mathbf{s} = 0 \qquad \text{on} \qquad \partial\Omega.$$

We note that this is the divergence-free constraint *strictly* imposed on the boundary $\partial\Omega$. This additional condition should not alter our solution drastically since it is exactly the constraint we are trying to satisfy everywhere in Ω. This condition, together with the natural boundary condition (3.28) results in the following Neumann boundary condition:

$$\frac{\partial}{\partial \eta}\mathbf{s} = 0 \qquad \text{on} \qquad \partial\Omega \tag{4.6}$$

where η is taken to be the direction normal to the surface $\partial\Omega$. Although (4.6) is a slight modification of the natural boundary condition (3.28), strictly speaking, this will result in a sub-optimal solution to the minimization problem (3.9) of Chapter 3. However, its attractiveness is in its simplicity lending itself to a standard matrix analysis and a wide variety of numerical algorithms.

For the Neumann boundary condition, the matrix equation (4.4), is of dimension $N_n \times N_n$ where $N_n = N_x N_y N_z$. As in the Dirichlet boundary condition case, we can represent this new equation as below:

$$\mathbf{A}_n\mathbf{x}_n = \mathbf{b}_n, \qquad \text{where} \quad \begin{cases} \mathbf{A}_n &= \mathbf{Q}_n + \gamma_1\, \mathbf{H}_{In}^T\, \mathbf{H}_{In} + \gamma_2\, \mathbf{H}_{Dn}^T\, \mathbf{H}_{Dn} \\ \mathbf{b}_n &= -\gamma_1\, \mathbf{H}_{In}^T\, \mathbf{f}_{in} \end{cases} \tag{4.7}$$

where the subscript $_n$ indicate a modified version of the matrices and the vectors of (4.4) to reflect the Neumann boundary condition (4.6).

4.2 Existence and Uniqueness of the Discrete Solution

Depending on the boundary condition imposed, the system of equations to be solved (4.4) may be significantly different. In fact, the Laplacian matrix assuming the Dirichlet boundary condition, $\mathbf{Q}_d$, is positive definite; however the Laplacian matrix assuming the Neumann boundary condition, $\mathbf{Q}_n$, is positive semi-definite. Therefore, we discuss the existence and the uniqueness of solutions to (4.5) and (4.7) separately. Also introduced in this section is the *mixed boundary condition* where we use the Dirichlet boundary condition on portions of the boundary $\partial\Omega$, where we know the value of $\mathbf{s}$, and the Neumann boundary condition on the remaining regions of $\partial\Omega$.

4.2.1 Dirichlet Boundary Condition

In this section, we show that $\mathbf{A}_d$ of equation (4.5) is non-singular and that it is in fact positive definite. The proof is by constructing the Laplacian matrix $\mathbf{Q}_d$ and showing that the eigenvalues of $\mathbf{Q}_d$ are strictly positive.

Stemming from the seven point 3-D Laplacian mask, and assuming the uniform sampling spacings to be $\Delta_x = \Delta_y = h_z\Delta_z$, we define the Laplacian of a discrete function g at grid (i,j,k) as:

$$\begin{aligned} -\nabla^2 g(i,j,k) &= (4+2h_z^2)\, g(i,j,k) \\ &\quad - g(i-1,j,k) - g(i+1,j,k) \\ &\quad - g(i,j-1,k) - g(i,j+1,k) \end{aligned} \tag{4.8}$$

$$- h_z^2[g(i,j,k-1) + g(i,j,k+1)]$$

$$(i,j,k) \in \{1,2,\ldots,N_x-2\} \times \{1,2,\ldots,N_y-2\} \times \{1,2,\ldots,N_z-2\}$$

where ∇^2 is the Laplacian operator, and we have normalized the distance $\Delta_x = 1$. In view of this, we define the following tri-diagonal matrix of dimension $(N_x - 2) \times (N_x - 2)$. This represents the terms $g(i,j,k), g(i-1,j,k)$ and $g(i+1,j,k)$ in (4.8).

$$\mathbf{B}_{d1} = \begin{bmatrix} 4+2h_z^2 & -1 & & & & & & \\ -1 & 4+2h_z^2 & -1 & & & & & \\ & -1 & 4+2h_z^2 & -1 & & & & \\ & & & \cdots & & & & \\ & & & -1 & 4+2h_z^2 & -1 & & \\ & & & & -1 & 4+2h_z^2 & -1 & \\ & & & & & -1 & 4+2h_z^2 \end{bmatrix} \tag{4.9}$$

The matrix $\mathbf{B}_{d1}$ has a special *toeplitz* structure that can be diagonalized by a sine transform whose eigenvalues are [49]

$$\lambda_l(\mathbf{B}_{d1}) = 4 + 2h_z^2 - 2\cos\frac{l\pi}{N_x - 1} \tag{4.10}$$

where $l = 1,2,\ldots,N_x - 2$.

For the terms $g(i,j-1,k)$ and $g(i,j+1,k)$ of (4.8), the block tri-diagonal

matrix of dimension $(N_x-2)(N_y-2)\times(N_x-2)(N_y-2)$ below is defined.

$$\mathbf{B}_{d2}=\begin{bmatrix} \mathbf{B}_{d1} & -\mathbf{I}_{d1} & & & & \\ -\mathbf{I}_{d1} & \mathbf{B}_{d1} & -\mathbf{I}_{d1} & & & \\ & \cdots & & & & \\ & & & -\mathbf{I}_{d1} & \mathbf{B}_{d1} & -\mathbf{I}_{d1} \\ & & & & -\mathbf{I}_{d1} & \mathbf{B}_{d1} \end{bmatrix} \tag{4.11}$$

where $\mathbf{I}_{d1}$ is the identity matrix whose dimensions are the same as $\mathbf{B}_{d1}$.

For the terms $g(i,j,k-1)$ and $g(i,j,k+1)$ of (4.8), we define the block tridiagonal matrix of dimension $N_d\times N_d$ where $N_d=(N_x-2)(N_y-2)(N_z-2)$.

$$\mathbf{B}_{d3}=\begin{bmatrix} \mathbf{B}_{d2} & -h_z^2\mathbf{I}_{d2} & & & & \\ -h_z^2\mathbf{I}_{d2} & \mathbf{B}_{d2} & -h_z^2\mathbf{I}_{d2} & & & \\ & \cdots & & & & \\ & & & -h_z^2\mathbf{I}_{d2} & \mathbf{B}_{d2} & -h_z^2\mathbf{I}_{d2} \\ & & & & -h_z^2\mathbf{I}_{d2} & \mathbf{B}_{d2} \end{bmatrix} \tag{4.12}$$

where $\mathbf{I}_{d2}$ is the identity matrix whose dimensions are the same as $\mathbf{B}_{d2}$.

Then finally, the following block diagonal matrix of dimension $3N_d\times 3N_d$ can

be shown to be $\mathbf{Q}_d$:

$$\mathbf{Q}_d = \begin{bmatrix} \mathbf{B}_{d3} & & \\ & \mathbf{B}_{d3} & \\ & & \mathbf{B}_{d3} \end{bmatrix} \tag{4.13}$$

Making repeated use of the method in [44, p. 148], together with the eigenvalues of $\mathbf{B}_{d1}$ (4.10), one can show that the eigenvalues of $\mathbf{Q}_d$ are

$$\lambda_{lmn}(\mathbf{Q}_d) = 4 + 2h_z^2 - 2\cos\frac{l\pi}{N_x - 1} - 2\cos\frac{m\pi}{N_y - 1} - 2h_z^2\cos\frac{n\pi}{N_z - 1} \tag{4.14}$$

with multiplicity three, where

$$\begin{aligned} l &= 1, 2, \ldots N_x - 2 \\ m &= 1, 2, \ldots N_y - 2 \\ n &= 1, 2, \ldots N_z - 2 \end{aligned}$$

We note that the eigenvalues $\lambda_{lmn}(\mathbf{Q}_d)$ are strictly positive and therefore $\mathbf{Q}_d$ must be positive definite. Since,

$$\mathbf{A}_d = \mathbf{Q}_d + \gamma_1\, \mathbf{H}_{Id}^T\, \mathbf{H}_{Id} + \gamma_2\, \mathbf{H}_{Dd}^T\, \mathbf{H}_{Dd},$$

it follows that $\mathbf{A}_d$ is positive definite as it is a sum of $\mathbf{Q}_d$, a positive definite matrix, and two non-negative matrices ($\gamma_1, \gamma_2 > 0$). Therefore, the discrete system of equations (4.5) has a unique solution.

4.2.2 Neumann Boundary Condition

In this section, we show that $\mathbf{A}_n$ of equation (4.7) is positive definite if the partial derivatives of the image f, $(\mathbf{f}_x, \mathbf{f}_y, \mathbf{f}_z)$, are linearly independent.

As in the previous section, we start by choosing the discrete Laplacian operator. Together with the conventional Laplacian mask (4.8) and the Neumann boundary condition (4.6), we infer the following Laplacian operator everywhere in Ω (including the boundary $\partial\Omega$).

$$\begin{aligned} -\nabla^2 g(i,j,k) &= (4+2h_z^2)\, g(i,j,k) \\ &\quad - g(l_M,j,k) - g(l_P,j,k) \\ &\quad - g(i,m_M,k) - g(i,m_P,k) \\ &\quad - h_z^2[g(i,j,n_M) + g(i,j,n_P)] \\ (i,j,k) &\in \{0,1,\ldots,N_x-1\} \times \{0,1,\ldots,N_y-1\} \times \{0,1,\ldots,N_z-1\} \end{aligned} \tag{4.15}$$

where

$$(l_M, l_P) = \begin{cases} (i, i+1), & i = 0 \\ (i-1, i+1), & i \in \{1,2,\ldots,N_x-2\} \\ (i-1, i) & i = N_x - 1 \end{cases}$$

$$(m_M, m_P) = \begin{cases} (j, j+1), & j = 0 \\ (j-1, j+1), & j \in \{1,2,\ldots,N_y-2\} \\ (j-1, j) & j = N_y - 1 \end{cases}$$

$$(n_M, n_P) = \begin{cases} (k, k+1), & k = 0 \\ (k-1, k+1), & k \in \{1, 2, \ldots, N_z - 2\} \\ (k-1, k) & k = N_z - 1 \end{cases}$$

Note that the Neumann boundary condition (4.6) is implicitly used here. For instance, at $x = 0$ (or $i = 0$), the Neumann boundary condition (4.6) supplies that $g_x = 0$. Assuming a backward difference scheme, we have $g(0, j, k) - g(-1, j, k) = 0$. Then, the Laplacian of g at $(0, j, k)$ can be defined by replacing $g(-1, j, k)$ with $g(0, j, k)$. At $x = T_x$ (or $i = N_x - 1$), we replace $g(N_x, j, k)$ with $g(N_x - 1, j, k)$ where we have assumed the forward difference scheme. It is similarly defined on other bounding planes. See [50], for instance.

In view of the discrete Laplacian operator above, we define the following tri-diagonal matrix of dimension $N_x \times N_x$. This represents the terms $g(i, j, k), g(i-1, j, k)$ and $g(i+1, j, k)$ in (4.15).

$$\mathbf{B}_{n1} = \begin{bmatrix} 4 + 2h_z^2 - 1 & -1 & & & & & & \\ -1 & 4 + 2h_z^2 & -1 & & & & & \\ & -1 & 4 + 2h_z^2 & -1 & & & & \\ & & & \cdots & & & & \\ & & & & -1 & 4 + 2h_z^2 & -1 & \\ & & & & & -1 & 4 + 2h_z^2 & -1 \\ & & & & & & -1 & 4 + 2h_z^2 - 1 \end{bmatrix} \tag{4.16}$$

The matrix $\mathbf{B}_{n1}$ has a special toeplitz structure that can be diagonalized by a cosine transform whose eigenvalues are [49]

$$\lambda_l(\mathbf{B}_{n1}) = 4 + 2h_z^2 - 2\cos\frac{(l-1)\pi}{N_x} \tag{4.17}$$

where $l = 1, 2, \ldots, N_x$.

For the terms $g(i, j-1, k)$ and $g(i, j+1, k)$ of (4.15), the block tri-diagonal matrix of dimension $N_xN_y \times N_xN_y$ below is defined.

$$\mathbf{B}_{n2} = \begin{bmatrix} \mathbf{B}_{n1} - \mathbf{I}_{n1} & -\mathbf{I}_{n1} & & & & \\ -\mathbf{I}_{n1} & \mathbf{B}_{n1} & -\mathbf{I}_{n1} & & & \\ & & \cdots & & & \\ & & & -\mathbf{I}_{n1} & \mathbf{B}_{n1} & -\mathbf{I}_{n1} \\ & & & & -\mathbf{I}_{n1} & \mathbf{B}_{n1} - \mathbf{I}_{n1} \end{bmatrix} \tag{4.18}$$

where $\mathbf{I}_{n1}$ is the identity matrix whose dimension are the same as $\mathbf{B}_{n1}$.

For the terms $g(i, j, k-1)$ and $g(i, j, k+1)$ of (4.15), we define the block

tri-diagonal matrix of dimension $N_n \times N_n$ where $N_n = N_x\, N_y\, N_z$.

$$\mathbf{B}_{n3} = \begin{bmatrix} \mathbf{B}_{n2} - h_z^2\mathbf{I}_{n2} & -h_z^2\mathbf{I}_{n2} & & & & \\ -h_z^2\mathbf{I}_{n2} & \mathbf{B}_{n2} & -h_z^2\mathbf{I}_{n2} & & & \\ & & \cdots & & & \\ & & & -h_z^2\mathbf{I}_{n2} & \mathbf{B}_{n2} & -h_z^2\mathbf{I}_{n2} \\ & & & & -h_z^2\mathbf{I}_{n2} & \mathbf{B}_{n2} - h_z^2\mathbf{I}_{n2} \end{bmatrix} \tag{4.19}$$

where $\mathbf{I}_{n2}$ is the identity matrix whose dimension are the same as $\mathbf{B}_{n2}$.

Then finally, the following block diagonal matrix of the dimension $3N_n \times 3N_n$ can be shown to be $\mathbf{Q}_n$:

$$\mathbf{Q}_n = \begin{bmatrix} \mathbf{B}_{n3} & & \\ & \mathbf{B}_{n3} & \\ & & \mathbf{B}_{n3} \end{bmatrix} \tag{4.20}$$

Making repeated use of the method in [44, p. 148], together with the eigenvalues of $\mathbf{B}_{n1}$ (4.17), one can show that the eigenvalues of $\mathbf{Q}_n$ are

$$\lambda_{lmn}(\mathbf{Q}_n) = 4 + 2h_z^2 - 2\cos\frac{(l-1)\pi}{N_x} - 2\cos\frac{(m-1)\pi}{N_y} - 2h_z^2\cos\frac{(n-1)\pi}{N_z} \tag{4.21}$$

with multiplicity three, where

$$\begin{aligned} l &= 1, 2, \ldots N_x \\ m &= 1, 2, \ldots N_y \end{aligned}$$

$$n = 1, 2, \ldots N_z$$

We note that the eigenvalues $\lambda_{lmn}(\mathbf{Q}_n)$ are positive except for λ_{111}, which is zero. Therefore $\mathbf{Q}_n$ is positive semi-definite. The eigenvector corresponding to the zero eigenvalue of $\mathbf{B}_{n3}$ is the first column of the cosine transform matrix which is a constant vector (the DC component).

In view of the above, the null space of $\mathbf{Q}_n$ can be shown to be

$$\mathcal{N}(\mathbf{Q}_n) = \left\{ a \begin{bmatrix} \mathbf{u} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + b \begin{bmatrix} \mathbf{0} \\ \mathbf{v} \\ \mathbf{0} \end{bmatrix} + c \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{w} \end{bmatrix} : \begin{array}{l} a, b, c \in \Re, \text{ and} \\ \mathbf{u} \in \Re^{N_x}, \mathbf{v} \in \Re^{N_y}, \mathbf{w} \in \Re^{N_z} \\ \quad \text{are all constant vectors} \end{array} \right\}.$$

Let

$$\mathbf{x}_0 = \begin{bmatrix} \mathbf{u}_0 \\ \mathbf{v}_0 \\ \mathbf{w}_0 \end{bmatrix} \in \mathcal{N}(\mathbf{Q}_n).$$

We note that

$$\mathbf{x}_0^T \mathbf{H}_{In}^T \mathbf{H}_{In} \mathbf{x}_0 = \| \operatorname{diag}(\mathbf{f}_x)\mathbf{u}_0 + \operatorname{diag}(\mathbf{f}_y)\mathbf{v}_0 + \operatorname{diag}(\mathbf{f}_z)\mathbf{w}_0 \|_2^2 \geq 0$$

where $\| \cdot \|_2$ is the usual Euclidean norm. The above can be zero only if $\mathbf{f}_x, \mathbf{f}_y$ and $\mathbf{f}_z$ are linearly dependent.

Therefore, assuming that $\mathbf{f}_x, \mathbf{f}_y$ and $\mathbf{f}_z$ are linearly independent, it follows that $\mathbf{x}^T \mathbf{H}_{In}^T \mathbf{H}_{In} \mathbf{x} \neq 0$ for all $\mathbf{x}$. Therefore, in this case

$$\mathbf{A}_n = \mathbf{Q}_n + \gamma_1 \mathbf{H}_{In}^T \mathbf{H}_{In} + \gamma_2 \mathbf{H}_{Dn}^T \mathbf{H}_{Dn}$$

is positive definite as $\gamma_1, \gamma_2 > 0$. Note the resemblance to the condition provided for the continuous case (see Theorem 3.7, Chapter 3). We repeat that this condition is generally satisfied by all images other than those that are carefully constructed.

Under this assumption (linear independence of $\mathbf{f}_x, \mathbf{f}_y$ and $\mathbf{f}_z$), the discrete set of equations (4.7) has a unique solution.

4.2.3 Mixed Boundary Condition

In some applications, we may have the values of $\mathbf{s}$ on parts of the boundary $\partial\Omega$. For instance, we may know that the imaged medium is contained within the bounding planes that are parallel to the z axis, i.e., at $x = 0, T_x$ and $y = 0, T_y$. In this case, the motion would certainly be zero on these four bounding planes and we would want to use the Dirichlet boundary condition here. At $z = 0, T_z$ we would still need to use the Neumann boundary condition.

In this case, the Laplacian matrix $\mathbf{Q}_m$ of (4.4) (subscript $_m$ for *mixed*) would have the form

$$\mathbf{Q}_m = \begin{bmatrix} \mathbf{B}_{m3} & & \\ & \mathbf{B}_{m3} & \\ & & \mathbf{B}_{m3} \end{bmatrix}. \tag{4.22}$$

which is of dimension $3N_m \times N_m$ where $N_m = N_x N_y (N_z - 2)$. The block tri-diagonal matrix $\mathbf{B}_{m3}$ of dimension $N_m \times N_m$ is shown below

$$\mathbf{B}_{m3} = \begin{bmatrix} \mathbf{B}_{d2} - h_z^2\mathbf{I}_{d2} & -h_z^2\mathbf{I}_{d2} & & & & \\ -h_z^2\mathbf{I}_{d2} & \mathbf{B}_{d2} & -h_z^2\mathbf{I}_{d2} & & & \\ & \cdots & & & & \\ & & & -h_z^2\mathbf{I}_{d2} & \mathbf{B}_{d2} & -h_z^2\mathbf{I}_{d2} \\ & & & & -h_z^2\mathbf{I}_{d2} & \mathbf{B}_{d2} - h_z^2\mathbf{I}_{d2} \end{bmatrix} \tag{4.23}$$

where $\mathbf{I}_{d2}$ and $\mathbf{B}_{d2}$ are defined as before.

Then, making repeated use of the method in [44, p. 148], together with the eigenvalues of $\mathbf{B}_{d1}$ (4.10) and $\mathbf{B}_{n1}$ (4.17), one can show that the eigenvalues of $\mathbf{Q}_m$ are

$$\lambda_{lmn}(\mathbf{Q}_m) = 4 + 2h_z^2 - 2\cos\frac{l\pi}{(N_x - 1)} - 2\cos\frac{m\pi}{(N_y - 1)} - 2h_z^2\cos\frac{(n-1)\pi}{N_z} \tag{4.24}$$

with multiplicity three, where

$$\begin{aligned} l &= 1, 2, \ldots N_x - 2 \\ m &= 1, 2, \ldots N_y - 2 \\ n &= 1, 2, \ldots N_z \end{aligned}$$

We note that the eigenvalues of $\mathbf{Q}_m$ are strictly positive, and therefore $\mathbf{Q}_m$ must be positive definite. Therefore the system matrix $\mathbf{A}$ of (4.4) is positive definite

for the mixed boundary condition.

In fact, as long as we are imposing the Dirichlet boundary condition on one of the six bounding planes, it is simple to show that the system matrix **A** is positive definite using the technique above.

4.3 Numerical Algorithms

The numerical solution to the linear system of equations (4.4) for Dirichlet, Neumann, or mixed boundary conditions is not trivial. For a typical 3-D medical image of 256 × 256×(*number of slices*), the dimension of the system matrix **A** would be prohibitively large. Therefore, a direct inversion of this matrix is difficult at best. (Although we have computed the eigenvalues of the Laplacian matrix **Q** we cannot compute the eigenvalues of a large typical **A**.) The consolation is that the matrix is sparse. We need an algorithm that is suitable for large but sparse matrices. For this purpose, we may use a variety of conjugate direction algorithms [51]. An advanced form of these algorithms that has a simple implementation approach as well as a favorable convergence property, is the *conjugate gradient algorithm.* This algorithm is reviewed in the next section

4.3.1 Conjugate Gradient Algorithm

The conjugate gradient algorithm was invented to minimize the general quadratic objective shown below:

$$\tfrac{1}{2}(\mathbf{x}, \mathbf{A}\mathbf{x}) - (\mathbf{b}, \mathbf{x}) \tag{4.25}$$

where $\mathbf{A}$ is symmetric and positive definite, and $(\cdot,\cdot)$ denote the Euclidean inner product. By taking the derivative of (4.25) with respect to $\mathbf{x}$ and setting it to zero, it is easy to see that this minimization problem is equivalent to solving the system of equations (4.4) repeated below:

$$\mathbf{Ax} = \mathbf{b}$$

Therefore, the conjugate gradient algorithm is directly applicable to our problem.

The algorithm is iterative in nature and it can be summarized as below using the notation above.

- Choose an arbitrary $\mathbf{x}^{(0)}$, usually $\mathbf{x}^{(0)} = \mathbf{0}$
- $\mathbf{p}^{(0)} = \mathbf{r}^{(0)} = \mathbf{b}^T - \mathbf{A}\mathbf{x}^{(0)}$
- Loop over $k = 0, 1, 2, \ldots$

  1. Compute $\mathbf{A}\mathbf{p}^{(k)}$
  2. $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \sigma_{\mathbf{x}}^{(k)}\mathbf{p}^{(k)}$
  3. $\mathbf{p}^{(k)} = \begin{cases} \mathbf{r}^{(0)}, & k = 0 \\ \mathbf{r}^{(k)} + \sigma_{\mathbf{p}}^{(k)}\mathbf{p}^{(k-1)}, & k = 1, 2, \ldots \end{cases}$
  4. $\mathbf{r}^{(k)} = \mathbf{r}^{(k-1)} - \sigma_{\mathbf{x}}^{(k-1)}\mathbf{A}\mathbf{p}^{(k-1)}$

- EndLoop if the residual $(\mathbf{r}^{(k)}, \mathbf{r}^{(k)})$ is sufficiently small

We choose $\sigma_{\mathbf{p}}^{(k)}$ so that $\mathbf{p}^{(n)}$ is A-conjugate to $\mathbf{p}^{(n-1)}$, i.e.,

$$(\mathbf{p}^{(n)}, \mathbf{A}\mathbf{p}^{(n-1)}) = 0$$

and choose $\sigma_{\mathbf{x}}^{(k)}$ to minimize

$$\tfrac{1}{2}(\mathbf{x}^{(n+1)}, \mathbf{A}\mathbf{x}^{(n+1)}) - (\mathbf{b}, \mathbf{x}^{(n+1)}).$$

After some algebraic manipulations, it can be shown that

$$\begin{aligned} \sigma_{\mathbf{x}}^{(k)} &= \frac{(\mathbf{r}^{(k)}, \mathbf{r}^{(k)})}{(\mathbf{p}^{(k)}, \mathbf{A}\mathbf{p}^{(k)})} \\ \sigma_{\mathbf{p}}^{(k)} &= \frac{(\mathbf{r}^{(k)}, \mathbf{r}^{(k)})}{(\mathbf{r}^{(k-1)}, \mathbf{r}^{(k-1)})} \end{aligned}$$

Note that this algorithm requires only one matrix multiplication per iteration.

For minimizing the penalty $e(\mathbf{s})$, the conjugate gradient algorithm as shown above can directly be implemented. Section 4.4 presents the results obtained by the conjugate gradient implementation of the 3-D motion recovery problem.

4.3.2 Preconditioned Conjugate Gradient Algorithm

There exists a class of preconditioners to improve the convergence properties of the conjugate gradient algorithm. An important source of the preconditioners is a class of basic iterative methods including RF, Jacobi, Gauss-Seidel, SOR and SSOR methods [52]. The goal of these basic methods is to solve the system of linear equations

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{4.26}$$

and can be described by the following iterative formula:

$$\mathbf{x}^{(n+1)} = \mathbf{G}\mathbf{x}^{(n)} + \mathbf{k} \tag{4.27}$$

where $\mathbf{G}$ (the iteration matrix that depends on the particular method in use), and $\mathbf{k}$ are:

$$\begin{aligned} \mathbf{G} &= \mathbf{I} - \mathbf{S}^{-1}\mathbf{A} \\ \mathbf{k} &= \mathbf{S}^{-1}\mathbf{b} \end{aligned} \tag{4.28}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{S}$ is the *splitting matrix.*

Given (4.28) with some invertible splitting matrix $\mathbf{S}$ and assuming $\mathbf{A}$ to be non-singular, the solution of the related system

$$(\mathbf{I} - \mathbf{G})\mathbf{x} = \mathbf{k}$$

is equivalent to the solution of (4.26). In this case, the iterative method (4.27) is said to be *completely consistent.*

The iterative method (4.27) is *symmetrizable* if there exists some non-singular matrix $\mathbf{W}$ such that the matrix $\mathbf{W}(\mathbf{I} - \mathbf{G})\mathbf{W}^{-1}$ is symmetric positive definite (SPD). Then, the method may be accelerated via the conjugate gradient algorithm of the previous section. We note in passing that symmetrizable basic iterative methods include RF, Jacobi, and SSOR methods.

With a symmetrization matrix, $\mathbf{W}$, instead of solving $\mathbf{Ax} = \mathbf{b}$ we solve the following:

$$\hat{\mathbf{A}}\hat{\mathbf{x}} = \hat{\mathbf{b}}$$

where

$$\hat{\mathbf{A}} = \mathbf{W}\mathbf{S}^{-1}\mathbf{A}\mathbf{W}^{-1}$$

$$\hat{\mathbf{x}} = \mathbf{W}\mathbf{x}$$

$$\hat{\mathbf{b}} = \mathbf{W}\mathbf{S}^{-1}\mathbf{b}$$

To solve this system, we may use the conjugate gradient algorithm of the previous section where we replace all variables with the *hat* ($\hat{}$) variables above. We would be working in the *transformed* space in this case. However, this adds unnecessary complications and results in additional multiplications/additions.

It is possible to derive a similar algorithm and run the algorithm in the *untransformed* space. The derivation is similar to the derivation of the conjugate gradient algorithm of the previous section. The preconditioned conjugate gradient algorithm (or the *conjugate gradient acceleration* of the basic iterative method) can be stated as below:

- Choose an arbitrary $\mathbf{x}^{(0)}$, usually $\mathbf{x}^{(0)} = \mathbf{0}$
- $\mathbf{p}^{(0)} = \mathbf{r}^{(0)} = \mathbf{b}^T - \mathbf{A}\mathbf{x}^{(0)}$
- Loop over $k = 0, 1, 2, \ldots$

  1. Compute $\mathbf{A}\mathbf{p}^{(k)}$
  2. $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \sigma_{\mathbf{x}}^{(k)}\mathbf{p}^{(k)}$
  3. $\mathbf{p}^{(k)} = \begin{cases} \mathbf{r}^{(0)}, & k = 0 \\ \mathbf{r}^{(k)} + \sigma_{\mathbf{p}}^{(k)}\mathbf{p}^{(k-1)}, & k = 1, 2, \ldots \end{cases}$
  4. $\mathbf{r}^{(k)} = \mathbf{r}^{(k-1)} - \sigma_{\mathbf{x}}^{(k-1)}(\mathbf{I} - \mathbf{G})\mathbf{p}^{(k-1)}$

- EndLoop if the residual $(\mathbf{r}^{(k)}, \mathbf{r}^{(k)})$ is sufficiently small

where

$$\sigma_{\mathbf{x}}^{(k)} = \frac{(\mathbf{Wr}^{(k)}, \mathbf{Wr}^{(k)})}{(\mathbf{Wp}^{(k)}, \mathbf{W}(\mathbf{I}-\mathbf{G})\mathbf{p}^{(k)})}$$
$$\sigma_{\mathbf{p}}^{(k)} = \frac{(\mathbf{Wr}^{(k)}, \mathbf{Wr}^{(k)})}{(\mathbf{Wr}^{(k-1)}, \mathbf{Wr}^{(k-1)})}$$

We note that the conjugate gradient algorithm of the previous section can be viewed as the conjugate gradient acceleration of the RF method since $\mathbf{G}_{RF} = (\mathbf{I}-\mathbf{A})$.

One particular preconditioner that we have used with some success is the conjugate gradient acceleration of the Jacobi method. We describe this method below.

We write the system matrix $\mathbf{A}$ as

$$\mathbf{A} = \mathbf{D} + \mathbf{C}_L + \mathbf{C}_U \tag{4.29}$$

where

$\mathbf{D}$ = a diagonal matrix with diagonal elements of $\mathbf{A}$

$\mathbf{C}_L$ = a lower triangular matrix with lower triangular elements of $\mathbf{A}$

$\mathbf{C}_U$ = an upper triangular matrix with upper triangular elements of $\mathbf{A}$

Then,

$$\mathbf{G}_{Jacobi} = \mathbf{D}^{-1}(-\mathbf{C}_L - \mathbf{C}_U)$$
$$= \mathbf{I} - \mathbf{S}^{-1}\mathbf{A}$$

Therefore, the splitting matrix $\mathbf{S}$, for the Jacobi method is $\mathbf{D}$. Moreover, the Jacobi method is symmetriazble with a symmetrization matrix $\mathbf{W} = \mathbf{A}^{\frac{1}{2}}$ or $\mathbf{W} = \mathbf{T}$ where $\mathbf{T}$ is any matrix such that $\mathbf{T}^T\mathbf{T} = \mathbf{D}$. For our particular problem, we chose to use the simplest symmetrization matrix

$$\mathbf{W} = \mathbf{D}^{\frac{1}{2}} \tag{4.30}$$

With the splitting matrix $\mathbf{S} = \mathbf{D}$ and the symmetrization matrix $\mathbf{W} = \mathbf{D}^{\frac{1}{2}}$, the preconditioned conjugate gradient acceleration algorithm can be simplified. The update multipliers can be computed as:

$$\begin{aligned} \sigma_{\mathbf{x}}^{(k)} &= \frac{(\mathbf{r}^{(k)}, \mathbf{D}\mathbf{r}^{(k)})}{(\mathbf{p}^{(k)}, \mathbf{A}\mathbf{p}^{(k)})} \\ \sigma_{\mathbf{p}}^{(k)} &= \frac{(\mathbf{r}^{(k)}, \mathbf{D}\mathbf{r}^{(k)})}{(\mathbf{r}^{(k-1)}, \mathbf{D}\mathbf{r}^{(k-1)})}. \end{aligned}$$

and the residual $\mathbf{r}$ can be computed as:

$$\mathbf{r}^{(k)} = \mathbf{r}^{(k-1)} - \sigma_{\mathbf{x}}^{(k-1)}\mathbf{D}^{-1}\mathbf{A}\mathbf{p}^{(k-1)}$$

As in the conjugate gradient algorithm, the preconditioned algorithm here requires one matrix multiplication per iteration. The inner products are slightly more complicated which increases the operation count. However, this increase is relatively small since the majority of the operation count stems from the matrix multiplication. Therefore, a direct comparison of the convergence rates of the two algorithms will show the advantage of one over the other. The comparison will be presented in the Section 4.5.

4.4 Results

In this section, results obtained by the conjugate gradient implementation of the algorithm of the previous section are presented. The 2-D version of the incompressibility constraint—the brightness constraint of optical flow—has been studied extensively; therefore, results obtained *(1)* with the incompressibility constraint only; and *(2)* with both incompressibility and divergence-free constraints are compared. It is demonstrated that for density images of moving incompressible objects, using the divergence-free constraint with the incompressibility constraint provides solutions consistent with our intuition of the motion of deforming objects. In all cases presented (except cine CT), the Dirichlet boundary condition was used since at the spatial boundaries of the images, the motion was known to be zero. For cine CT of a beating heart, the Dirichlet boundary condition was used for the four boundary planes normal to the x-y plane; and the Neumann boundary condition was used for the remaining planes, i.e., the first and last slices of the 3-D image (planes $k = 0$ and $k = N_z - 1$).

4.4.1 Simulated Images

All simulated images presented in this section were constructed so that the conservation of mass is obeyed. Regions of images were allowed to deform but they were not allowed to change in the area that they occupy in both time frames. Hence, these simulated images clearly satisfies the conservation of mass (2.7). The incompressibility constraint was imposed on the motion by maintaining a constant gray level for each fixed point of the image as the objects deformed. Thus, it is clear that the velocity field generated by the simulated images of Figures 4.1

and 4.6 were constructed to satisfy the incompressibility and the divergence-free constraints.

2-D images

For illustration purposes, 2-D examples are presented first. Experiment 1 of Figure 4.1 shows a 64×64 image sequence of a translating circle. The boundary of the outer circle (of radius 25 pixels) is fixed whereas the inner circle (of radius 13 pixels) translates one pixel down. A physical example of this type of motion is a situation in which the inner region is solid and the outer annulus is fluid. The image intensities represent the densities of these regions. As the inner circle translates, the outer region (the outer boundary is fixed) experiences motion such that along the left and right regions of the annulus, upward motion should prevail to evacuate (bottom) and replenish (top) the regions affected by the inner region's motion.

The incompressibility constraint alone does not give results that show such motion. Figure 4.2 (a) shows the result obtained using only the incompressibility constraint. Here, a general downward motion is shown which does not agree with the preceding arguments. When using both constraints, the motion within the annular region is seen to accommodate the motion of the inner circle as described above. This is clearly indicated in Figure 4.2 (b).

Experiment 2 of Figure 4.1 shows a 64×64 image sequence of a diagonally translating circle. The boundary of the outer circle is fixed as for the first experiment, and the inner circle translates 1 pixel diagonally. Both circles are of the same radius as in the first experiment. Figure 4.3 depicts similar results as in experiment 1.

Experiment 3 of Figure 4.1 shows a 64×64 image sequence of a deforming ellipse. The outer ellipse (a circle of radius 25) is fixed in both frames and the inner ellipse deforms from a major and minor axis of (13, 10) to (15, 130/15). These numbers were chosen to guarantee the conservation of mass—i.e. the equation of continuity (2.9). Again, the utility of the divergence-free constraint for density images is clearly illustrated in Figure 4.4.

These experiments in two dimensions clearly indicate the advantage of using the divergence-free constraint for density images.

In comparing the results of experiments 1 and 2 we found that the choice of the parameters γ_1 and γ_2 plays a role in the appearance of the estimated velocity field. As γ_1 was increased we observed the appearance of vortices in the estimated field. Conversely, as γ_1 is decreased, the field becomes over-smoothed. Figure 4.5 shows this strong dependency on γ_1. The solution appears to be less sensitive to the choice of γ_2. We conclude that if this method is to be used with confidence, the problem of selecting these parameters must be addressed. One can do this treating γ_1 and γ_2 as Lagrange multipliers [48, 53] which will be discussed in Chapter 5.

3-D images

Experiment 4 of Figure 4.6 shows a 16×16×8 image sequence of a translating ellipsoid. The outer ellipsoid is fixed in both frames and the inner ellipsoid translates down one voxel. As for the experiments in 2-D, the algorithm was performed with and without the divergence-free constraint. Figure 4.7 shows the two 3-D vector fields plotted as a function of 3-D space, projected onto a plane. Figure 4.7 (a)—without the divergence-free constraint—does indicate a general

downward motion. However, the motion deep within the inner ellipsoid is significantly smaller than it should be. Figure 4.7 (b) does not have this undesirable effect due to the divergence-free constraint.

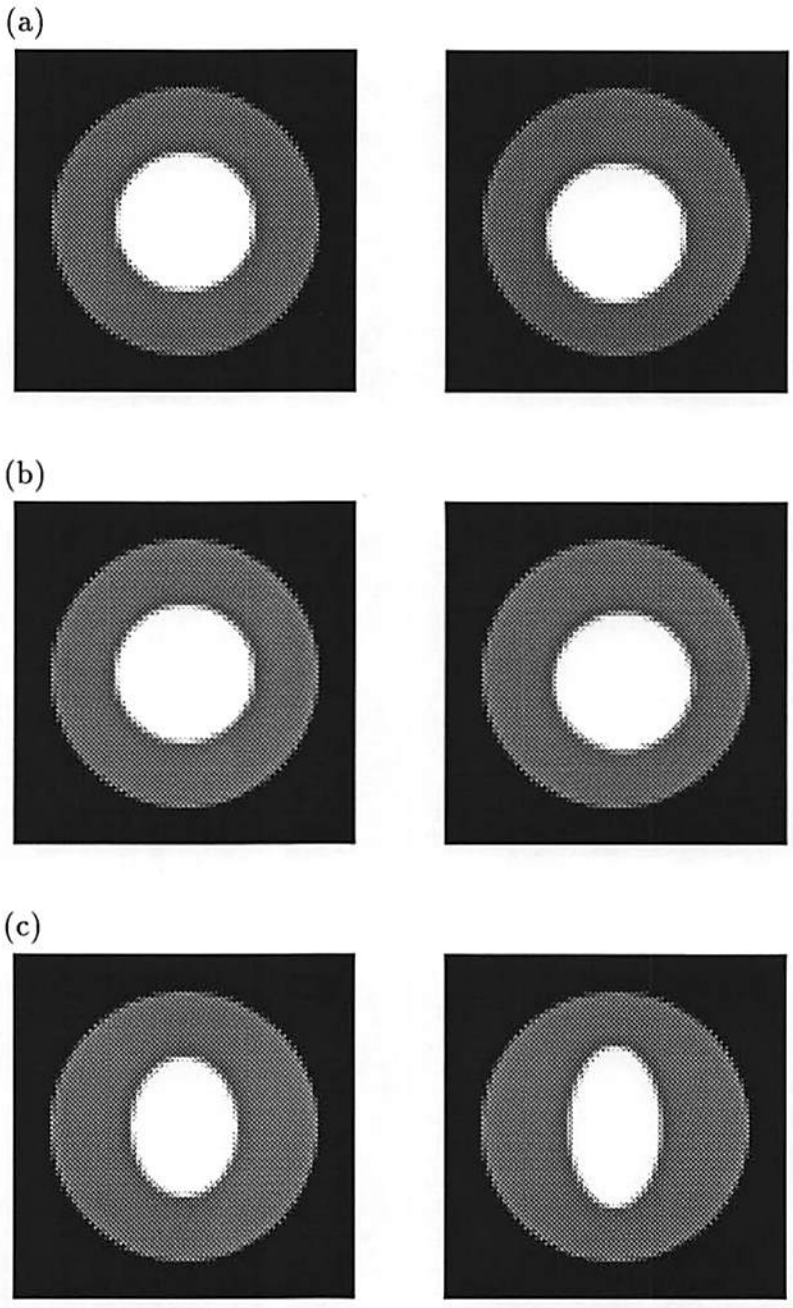
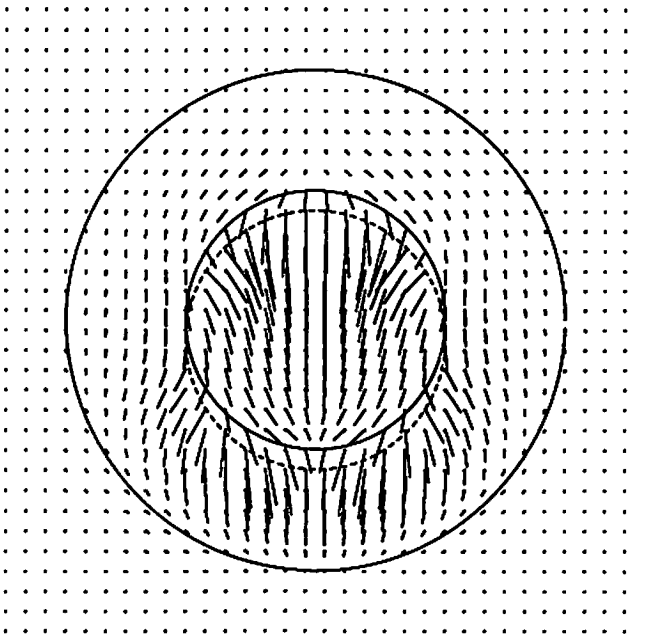
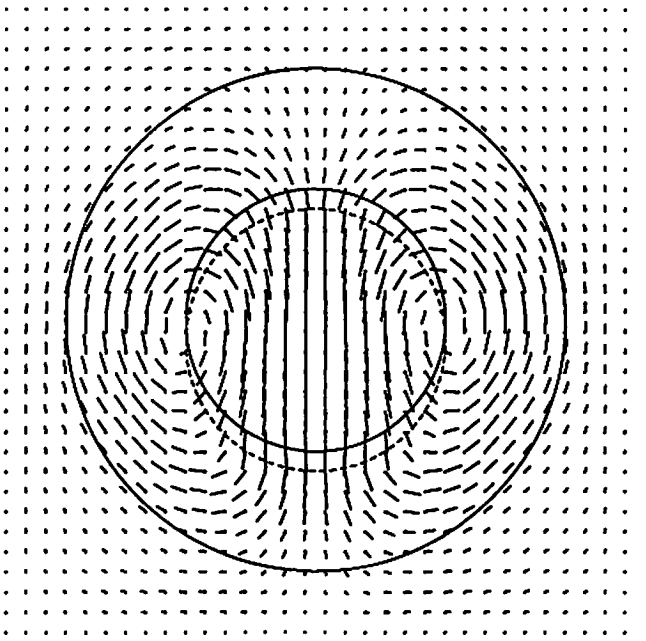


Figure 4.1: Simulated images for experiments 1, 2 and 3. (a) Experiment 1: Vertically translating circle, time frames 1 and 2. (b) Experiment 2: Diagonally translating circle, time frames 1 and 2. (c) Experiment 3: Deforming ellipse, time frames 1 and 2. The boundary of the outer region is fixed and the motion arises from the inner regions

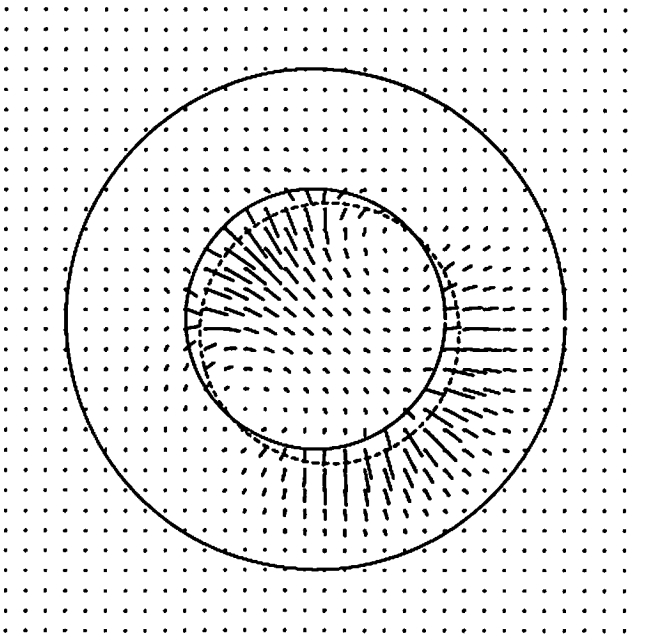


(a)

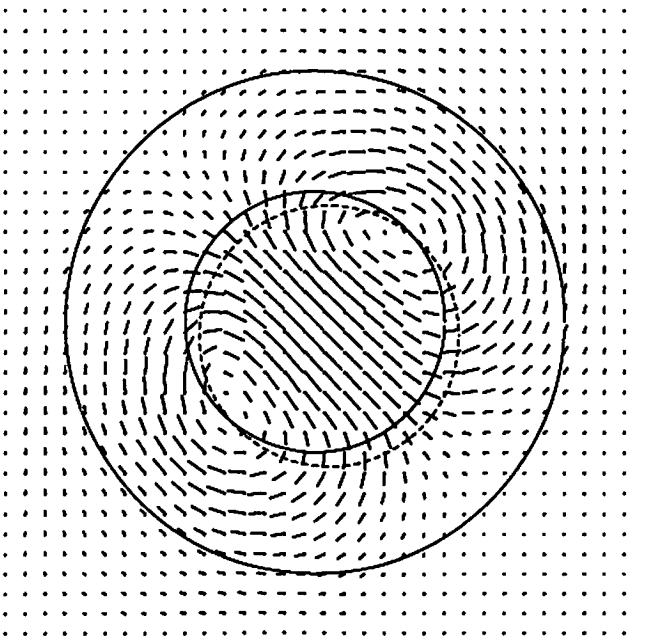


(b)

Figure 4.2: Results of experiment 1 with the boundary outlined. The dotted circle shows the inner circle's position in the second frame. (a) Incompressibility constraint only. (b) Incompressibility and divergence-free constraints.

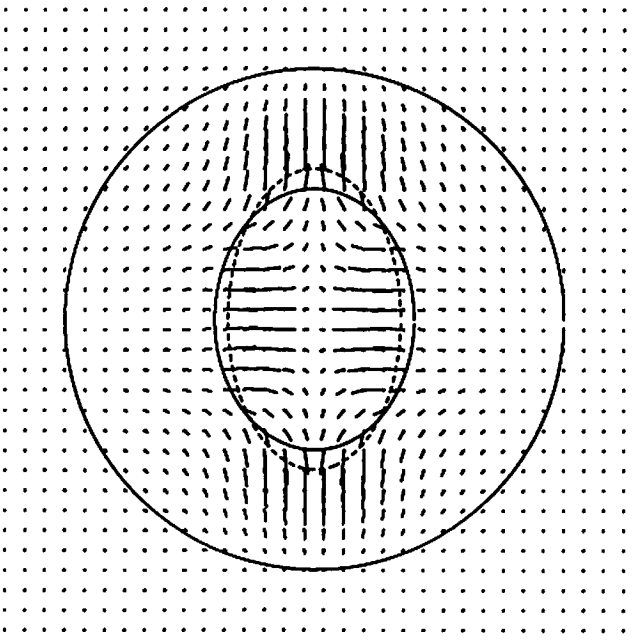


(a)

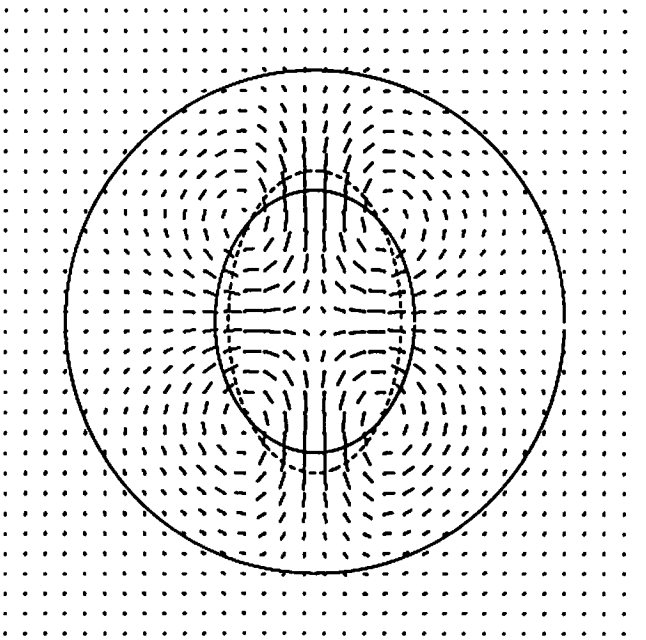


(b)

Figure 4.3: Results of experiment 2 with the boundary outlined. The dotted circle shows the inner circle's position in the second frame. (a) Incompressibility constraint only. (b) Incompressibility and divergence-free constraints.



(a)



(b)

Figure 4.4: Results of experiment 3 with the boundary outlined. The dotted ellipse shows the inner ellipse's position in the second frame. (a) Incompressibility constraint only. (b) Incompressibility and divergence-free constraints.

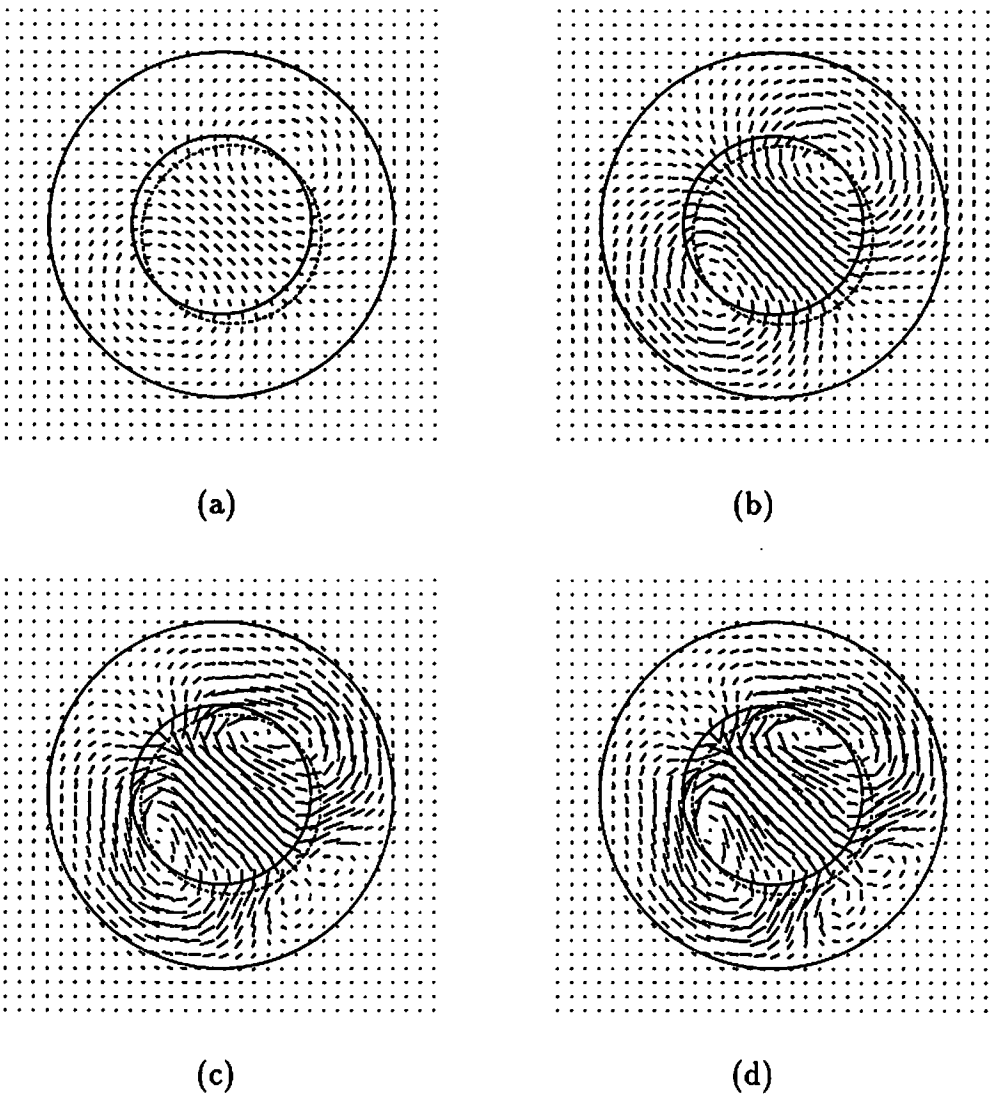


Figure 4.5: Results of experiment 2 with different values of γ_1; and $\gamma_2 = 250$. The dotted circle shows the inner circle's position in the second frame. (a) $\gamma_1 = .001$, (b) $\gamma_1 = .005$, (c) $\gamma_1 = .01$, (d) $\gamma_1 = .02$.

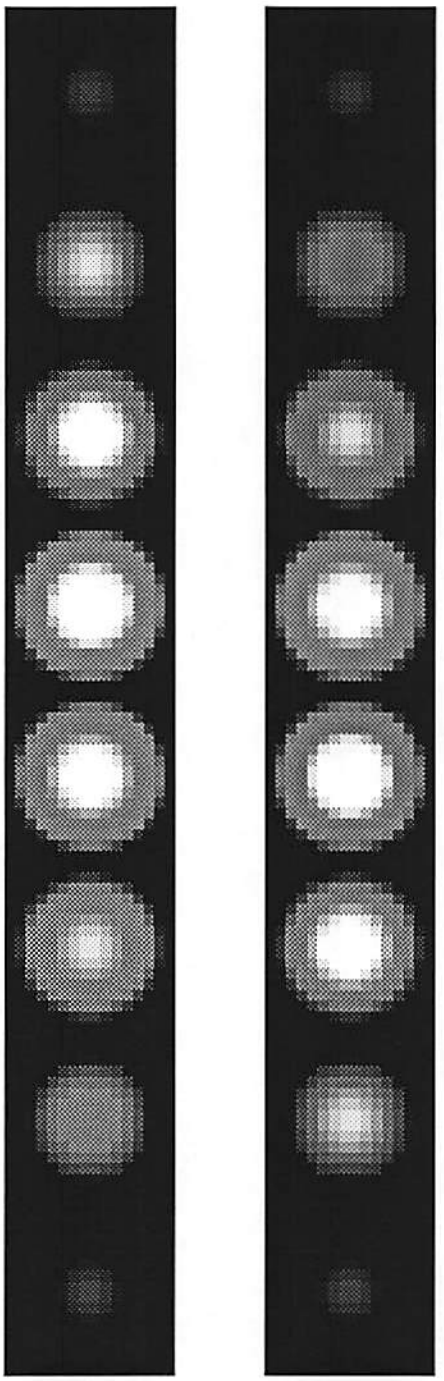
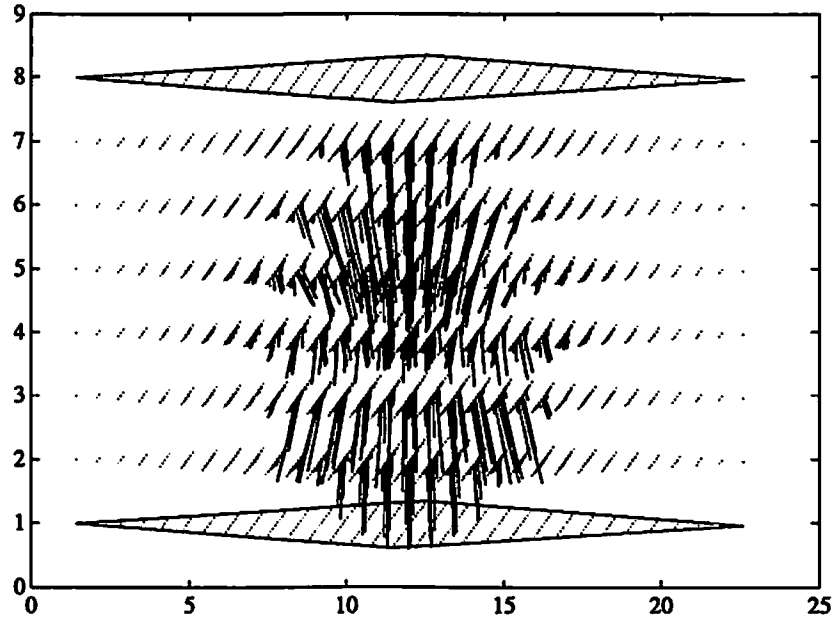
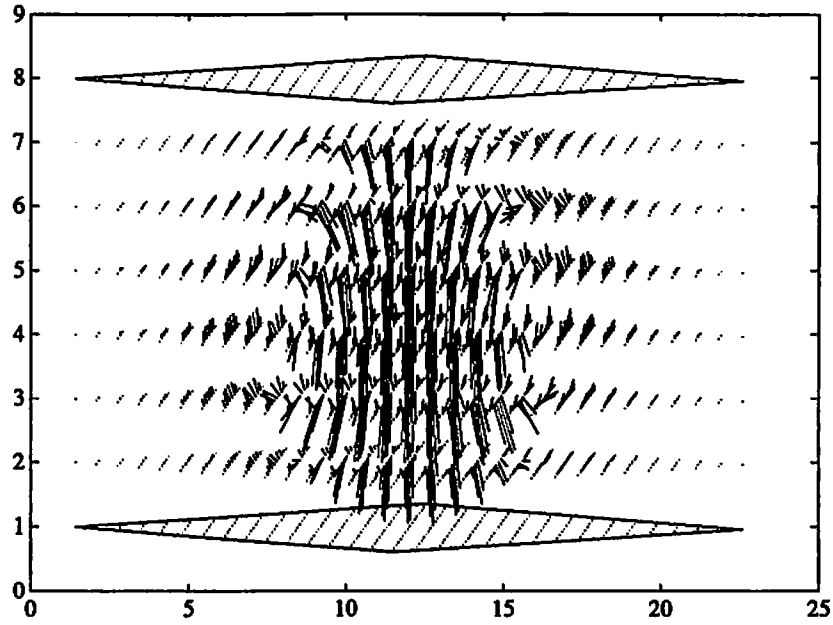


Figure 4.6: Simulated images for experiment 4: Vertically translating ellipsoid, time frames 1 and 2. The outer ellipsoid is fixed and the inner ellipsoid translates down one voxel.



(a)



(b)

Figure 4.7: Results of experiment 4. 3-D vector field as a function of a 3-D space is projected onto a plane. (a) Incompressibility constraint only. (b) Incompressibility and divergence-free constraints.

4.4.2 Cine CT of a beating heart

Images from the Imatron Ultrafast Scanner [1, 54] were used to evaluate the performance of the algorithm for real data. Low noise in the sequence of cardiac images provided by the Ultrafast Scanner was particularly attractive since the algorithm is based on partial derivatives of the given images. The details concerning the Ultrafast CT image acquisition can be found in [1, 54]. The images used in our study were acquired using a contrast agent introduced via intravenous injection. Provided (a) the contrast agent enters the imaged volume through a boundary at which the Neumann boundary condition is used, i.e., the plane $k = 0$ or $k = N_z - 1$, and (b) we assume that the mixing of the blood and contrast medium occurs outside of the heart, we would expect the internal velocity field to satisfy both the incompressibility constraint and the divergence-free constraint.

The Imatron data set used in this study was 15 time frames of 3-D images of size 256×256×8. Frame 1 represents the beginning of the R-wave. All slices were $8mm$ in thickness. These 3-D images were cut to reduce the computational burden to 128×128×8 with the heart at the center of the new images. The reduced images were further processed with a simple 2×2 averaging filter (in the x-y plane) followed by a 2:1 decimation to give 15 frames of 64×64×8 images. The algorithm was first performed on frames 1 and 2.

It is difficult to depict a 3-D velocity field as a function of 3-D space. For a simple motion, such as in experiment 4, the method used in Figure 4.7 proved useful; however, for complicated motion, these illustrations are difficult to interpret. We present a partial result here by overlaying the velocity vectors onto a plane of the heart image. A complete result can be presented in this manner for any plane desired. Figures 4.8 and 4.9 show all 8 slices of frame 1 overlayed with

the x and y components of the computed velocity. The velocity vectors shown here represent the displacement of individual mass from frame 1 to 2.

The process was repeated for all pairs of frames, (2,3), (3,4), ... , (15,1), so that the results may be viewed as a movie. Here, we present a portion of the movie. Figures 4.10 and 4.11 show levels 4 and 5 just before and after the left ventricular contraction. In both levels, frames 3 through 5 show the latter part of the systolic phase of the heart cycle. The velocity vectors in these frames show a strong contracting motion of the left ventricle. Frame 6 of both levels shows a near-zero velocity field about the left ventricle as this frame is approximately the end-systole. Frame 7 is the beginning of the diastole. The velocity field about the left ventricle in frames 7 and 8 indicates the expansion of the left ventricle.

The motion of the heart in one time frame is small since the Imatron Ultrafast Scanner provides up to 17 frames per heart beat. Careful examination of the images indicate that the algorithm provides reasonable estimates of the velocity field, at least visually.

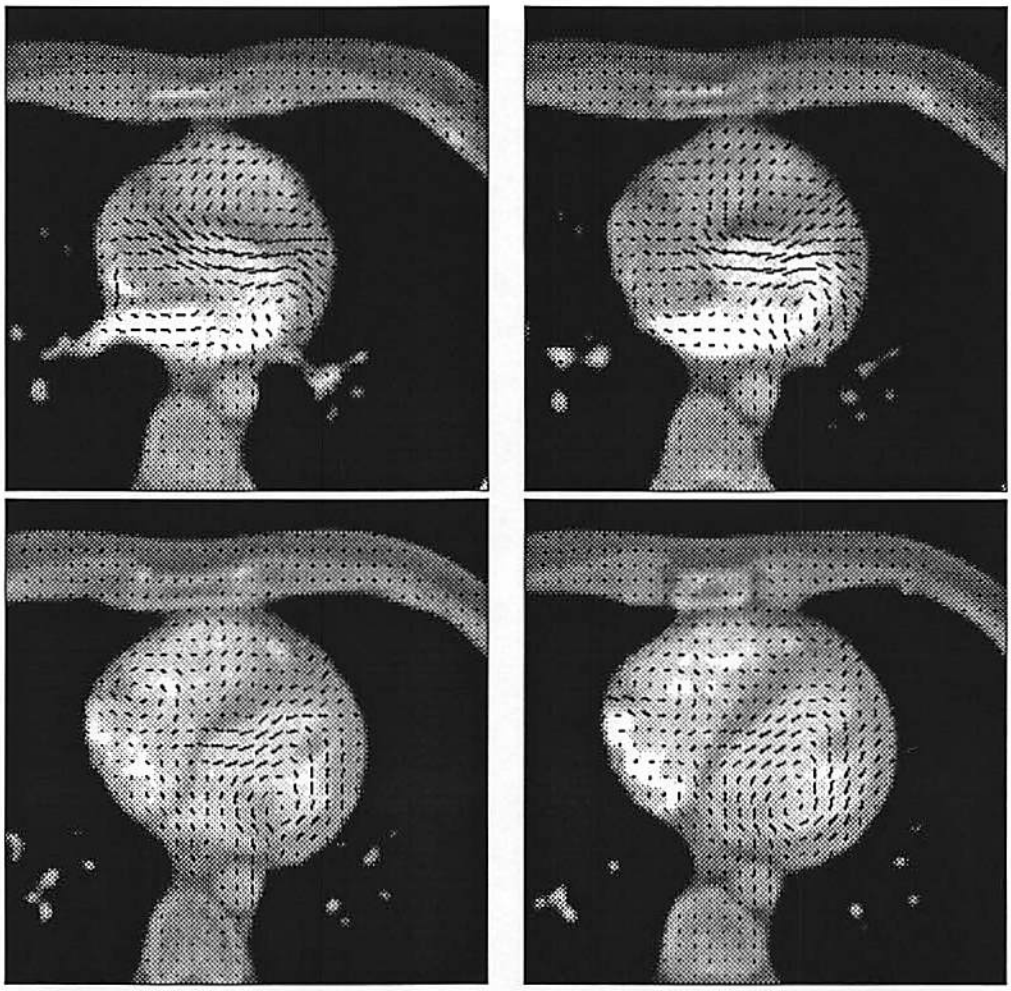


Figure 4.8: Cine CT of a beating heart overlayed with velocity field vectors. Levels 1 through 4, frame 1.

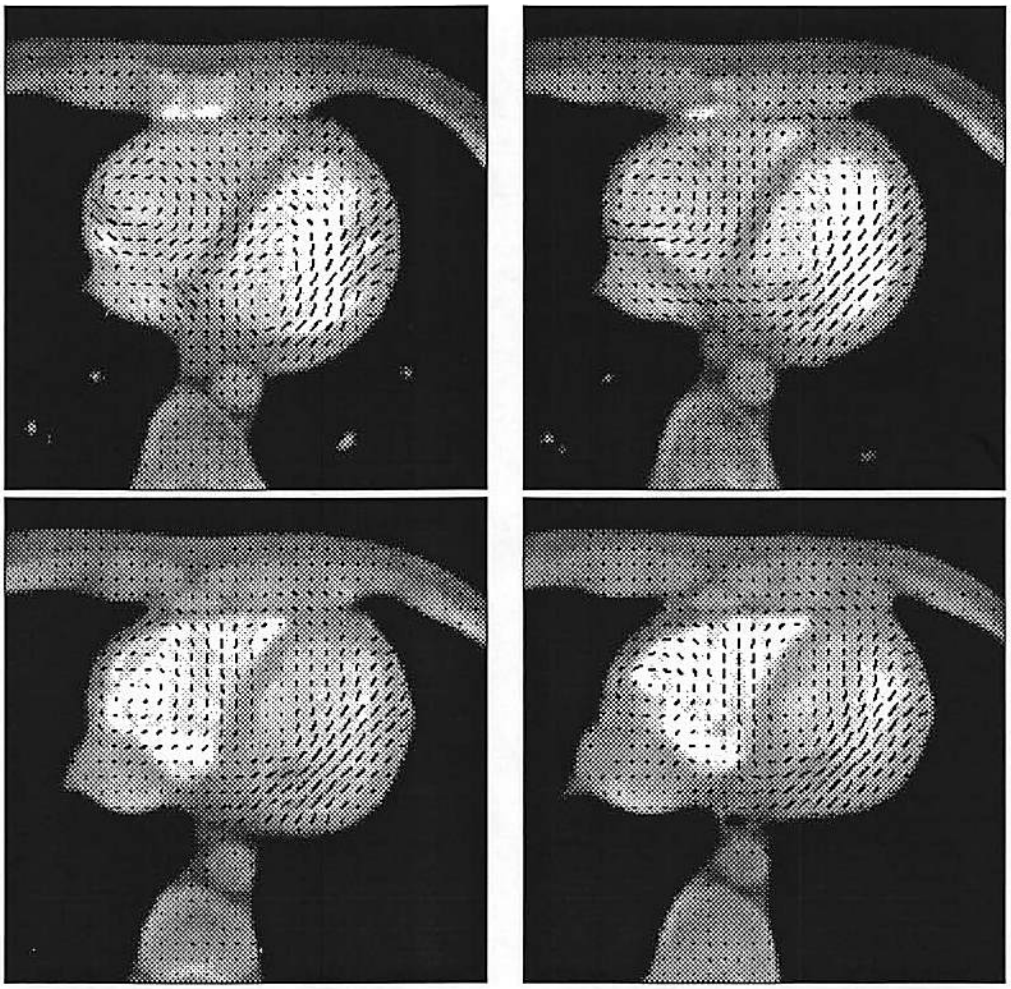


Figure 4.9: Cine CT of a beating heart overlayed with velocity field vectors. Levels 5 through 8, frame 1.

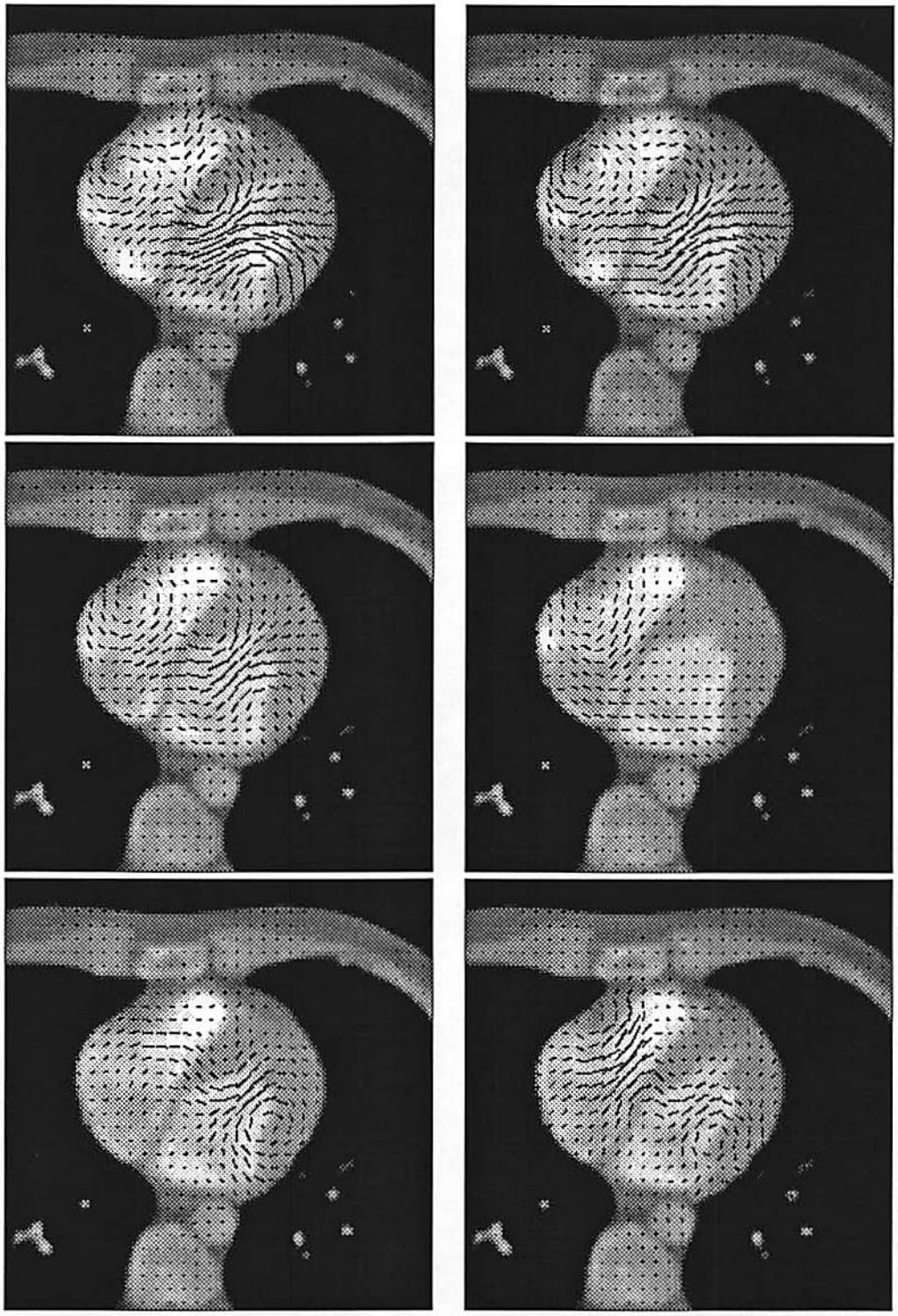


Figure 4.10: Cine CT of a beating heart overlayed with velocity field vectors. Level 4, frames 3 through 8.

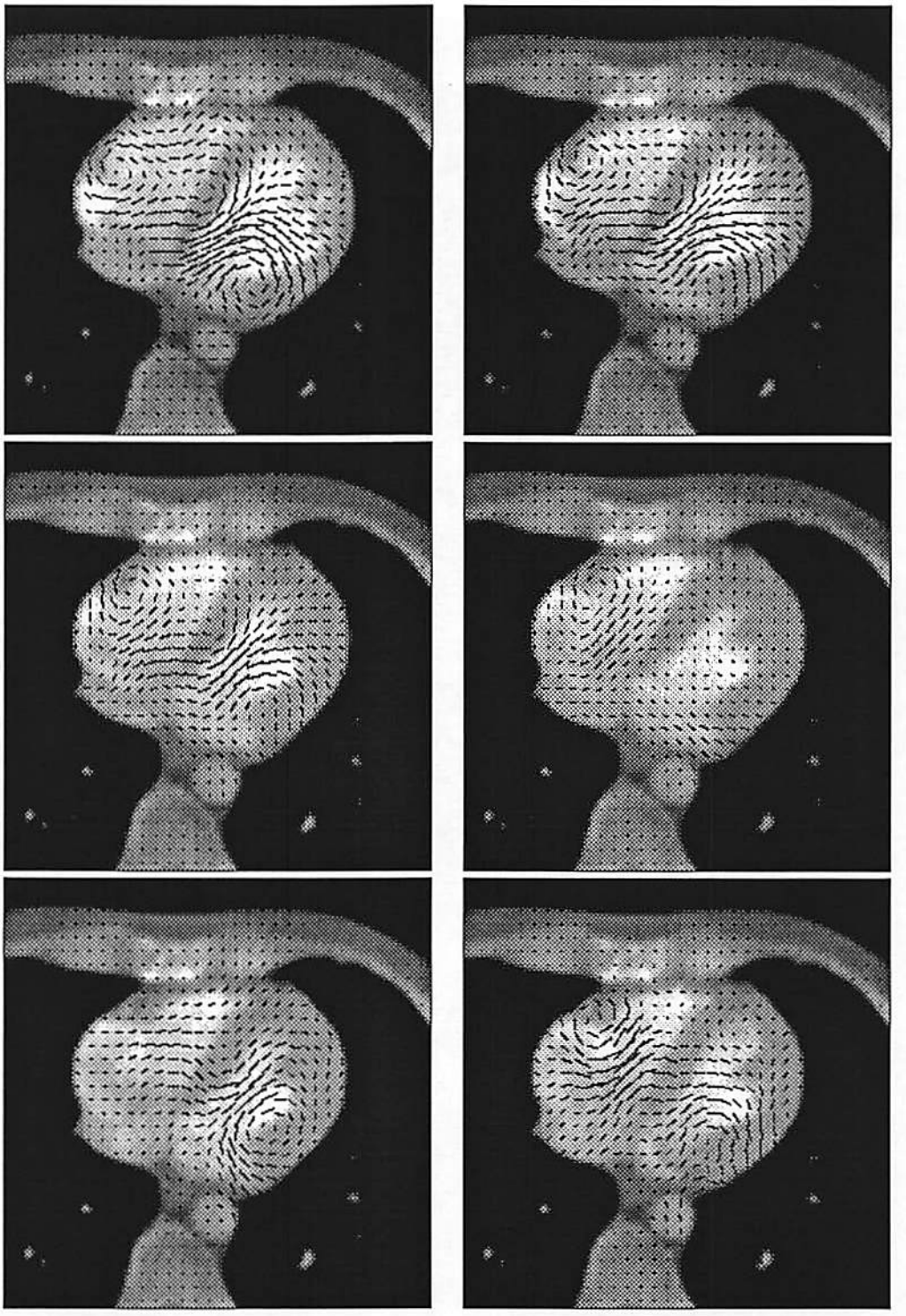


Figure 4.11: Cine CT of a beating heart overlayed with velocity field vectors. Level 5, frames 3 through 8.

4.5 Preconditioned Conjugate Gradient Algorithm: A Performance Study

In this section, we compare the convergence rates of the conjugate gradient algorithm of Section 4.3.1 and the preconditioned conjugate gradient algorithm of Section 4.3.2. The system matrix $\mathbf{A}$ of (4.4) is not trivial as it includes terms from the image f. We have not found a method to compare or even to bound the condition numbers of $\mathbf{A}$ and $\hat{\mathbf{A}} = \mathbf{W}\mathbf{S}^{-1}\mathbf{A}\mathbf{W}^{-1}$. We present an empirical comparison.

The number of additions per iteration is identical for both algorithms. However, each iteration of the preconditioned algorithm involves a few more multiplications. Table 4.1 shows the number of multiplications per iteration per pixel. Since both algorithms require one matrix multiplication per iteration, and most operations originate from the matrix multiply, the complexity of the two algorithms is highly similar. Therefore, a direct comparison of the convergence rates will establish the superiority of one over the other.

	Conjugate Gradient	Preconditioned C-G
matrix multiply	42	42
inner product	6	9
others	9	12
TOTAL	**57**	**63**

Table 4.1: Multiplication count for the conjugate gradient algorithm and the preconditioned conjugate gradient algorithm.

Both algorithms were run with the same 3-D images of a beating heart of the previous section (time frames 1 and 2). Figure 4.12 shows the convergence rates

for both algorithms. The plots show the residual $\mathbf{r}^{(k)}$ on log scale:

$$\log_{10} \frac{\| \mathbf{r}^{(k)} \|_2^2}{\| \mathbf{r}^{(0)} \|_2^2}$$

The solid line shows the residual for the conjugate gradient algorithm and the dotted line shows the residual for the preconditioned algorithm. Other time frames showed similar convergence rates.

It is difficult to assess the performance of a preconditioner especially for a complex system matrix $\mathbf{A}$. In this section, we have empirically demonstrated that for real images, the preconditioned algorithm converges to a within a given tolerance with fewer operations. For instance, for the normalized error of 1/100, i.e.,

$$\frac{\| \mathbf{r}^{(k)} \|_2^2}{\| \mathbf{r}^{(0)} \|_2^2} = \frac{1}{100},$$

the conjugate gradient algorithm takes approximately 30 iterations; however, the preconditioned conjugate gradient algorithm takes less than 20 iterations.

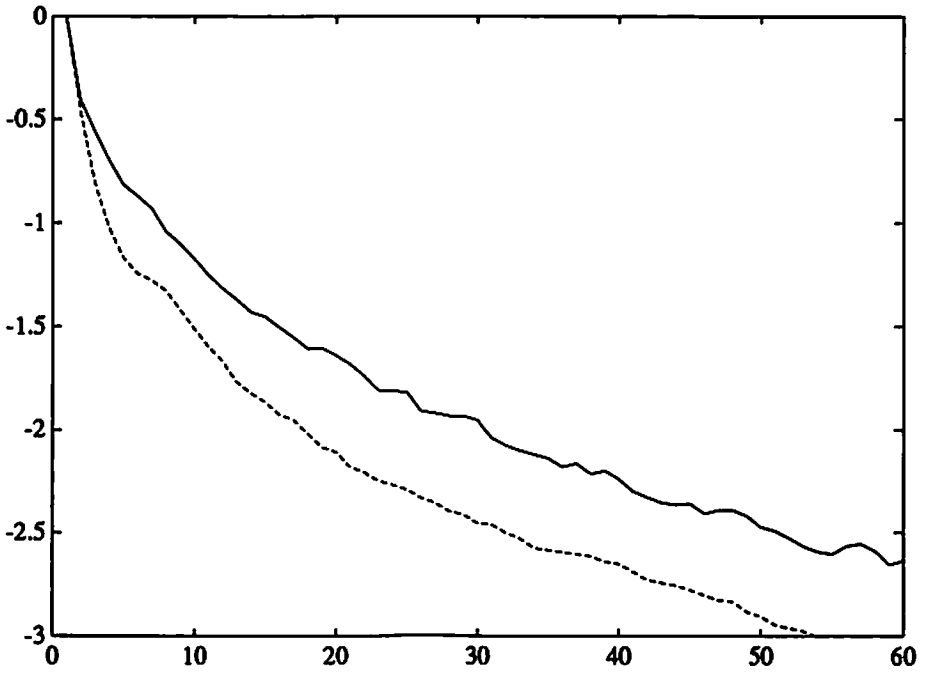


Figure 4.12: Convergence rates of the conjugate gradient algorithm and the preconditioned conjugate gradient algorithm. Solid line—conjugate gradient algorithm. Dotted line—preconditioned conjugate gradient algorithm.

Chapter 5

Lagrange Multiplier Method: Obtaining the Regularization Parameters

In Chapter 3, we have formulated and solved (in the continuous domain) the unconstrained minimization problem below:

$$\textit{minimize} \qquad e(\mathbf{s}) = e_S(\mathbf{s}) + \gamma_1\, e_I(\mathbf{s}) + \gamma_2\, e_D(\mathbf{s})$$

where

$$\begin{aligned} e_S(\mathbf{s}) &= \int_\Omega (u_x^2 + u_y^2 + u_z^2 + v_x^2 + v_y^2 + v_z^2 + w_x^2 + w_y^2 + w_z^2)\, d\Omega \\ e_I(\mathbf{s}) &= \int_\Omega (f_x u + f_y v + f_z w + f_t)^2\, d\Omega \\ e_D(\mathbf{s}) &= \int_\Omega (u_x + v_y + w_z)^2\, d\Omega \end{aligned}$$

A solution in the discrete domain was provided in Chapter 4 along with some numerical examples. However, the solution to the above minimization problem will be different for different values of γ_1 and γ_2. Therefore, as previously noted, the solution (to the original 3-D motion recovery problem) depends on the choice of these parameters. In this chapter we develop an algorithm to compute these parameters adaptively.

These parameters are sometimes referred to as the regularization parameters [3]. See Section 3.1.1 for a brief description of Tikhonov's regularization method for solving ill-posed problems.

A recent publication [55] dwells on obtaining this parameter for linear problems. The reported methods are applicable for image restoration problems where the image model is linear:

$$g = \mathbf{H}\rho + \epsilon \tag{5.1}$$

where g is the degraded image; $\mathbf{H}$, the blurring matrix; ρ, the original image; and ϵ, the additive noise. For our particular problem, $\mathbf{b} = \mathbf{A}\mathbf{x}$, where $\mathbf{x}$ is the desired velocity field vector. The noisy image data is embedded in both the matrix $\mathbf{A}$ and the vector $\mathbf{b}$. Therefore, the linear additive noise model (5.1) cannot be used here. We will describe methods that are suitable for our problem.

5.1 Determining One Parameter: Divergence-free Term

The noise is contained in the image data f and nowhere else. It is unlikely that the incompressibility constraint will be satisfied exactly, since it contains

terms dependent on the image f, i.e., (f_x, f_y, f_z, f_t); however, the divergence-free constraint should be satisfied exactly (except for the round-off noise during the computation). Then, perhaps, we should pose the following linearly constrained problem:

$$\begin{aligned} \textit{minimize} \quad & \int_\Omega \{(u_x^2 + u_y^2 + u_z^2 + v_x^2 + v_y^2 + v_z^2 + w_x^2 + w_y^2 + w_z^2) \\ & \qquad + \gamma_1 (f_x u + f_y v + f_z w + f_t)^2 \} \, d\Omega \\ \textit{subject to} \quad & u_x + v_y + w_z = 0 \end{aligned} \tag{5.2}$$

where γ_1 remains a positive constant as before. We present in detail, the solution to (5.2) in this section.

Let F be the integrand of the above integral. Then, from the theory of Lagrange optimization, the solution $\mathbf{s}_0$ must satisfy the following set of Euler-Lagrange equations with either the Dirichlet or the natural boundary conditions (see Appendix A.2).

$$\begin{aligned} 0 &= F_u - \frac{\partial}{\partial x}(F_{u_x} + \lambda) - \frac{\partial}{\partial y} F_{u_y} - \frac{\partial}{\partial z} F_{u_z} \\ 0 &= F_v - \frac{\partial}{\partial x} F_{v_x} - \frac{\partial}{\partial y}(F_{v_y} + \lambda) - \frac{\partial}{\partial z} F_{v_z} \\ 0 &= F_w - \frac{\partial}{\partial x} F_{w_x} - \frac{\partial}{\partial y} F_{w_y} - \frac{\partial}{\partial z}(F_{w_z} + \lambda) \end{aligned}$$

where $\lambda = \lambda(x, y, z)$ is the Lagrange multiplier for this problem. The evaluation of the above Euler-Lagrange equations results in the following set of PDEs.

$$\begin{aligned} \nabla^2 u &= \gamma_1 \, f_x (f_x u + f_y v + f_z w + f_t) - \tfrac{1}{2} \lambda_x \\ \nabla^2 v &= \gamma_1 \, f_y (f_x u + f_y v + f_z w + f_t) - \tfrac{1}{2} \lambda_y \end{aligned} \tag{5.3}$$

$$\nabla^2 w \;=\; \gamma_1 \, f_z(f_x u + f_y v + f_z w + f_t) \;-\; \tfrac{1}{2}\,\lambda_z$$

Using the notation of Chapter 4, discretization of the above PDEs yields

$$\begin{aligned} (\mathbf{Q} + \gamma_1 \mathbf{H}_I^T \mathbf{H}_I)\ \mathbf{x} + \tfrac{1}{2}\mathbf{H}_D^T\ \boldsymbol{\lambda} &= -\gamma_1 \mathbf{H}_I^T \mathbf{f}_t \\ \tfrac{1}{2}\mathbf{H}_D\ \mathbf{x} &= \mathbf{0} \end{aligned}$$

where the second equation is from the constraint $u_x + v_y + w_z = 0$. The matrices $\mathbf{H}_D, \mathbf{H}_I, \mathbf{Q}$ and $\mathbf{x}$ are defined as previously in Chapter 4 equation (4.3), and $\boldsymbol{\lambda}$ is the vector stacked version of $\lambda(x, y, z)$. The equations above may be written in the matrix form $\mathbf{By} = \mathbf{c}$ as:

$$\begin{bmatrix} \mathbf{Q} + \gamma_1 \mathbf{H}_I^T \mathbf{H}_I & \frac{1}{2}\mathbf{H}_D^T \\ \frac{1}{2}\mathbf{H}_D & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -\gamma_1 \mathbf{H}_I^T \mathbf{f}_t \\ \mathbf{0} \end{bmatrix} \tag{5.4}$$

The solution to the above matrix equation is the numerical solution to the linearly constrained minimization problem (5.2). The equation (5.4) is not trivial to solve since the system matrix $\mathbf{B}$ is indefinite. Various algorithms including all basic iterative methods (RF, Jacobi, Gauss-Seidel, etc) and the conjugate gradient algorithm cannot be used here since these algorithms are useful only for positive-definite systems.

There exists a class of algorithms to deal with such indefinite systems [56]. Luenberger's hyperbolic pairs method [57], which is a modified version of the conjugate gradient algorithm, was tried but the algorithm could not recover upon a singular direction [57].

We have also tried a simple method where we solve the equation $\mathbf{B}^2\mathbf{y} = \mathbf{Bc}$

instead, by the conjugate gradient algorithm. Here, convergence was difficult to obtain as the condition number is made worse by *1)* using $\mathbf{B}^2$ instead of $\mathbf{B}$, *2)* increasing the dimension of the problem by a factor 4/3 (we have to compute the Lagrange multiplier as well as the velocity field vector). Also, for the same reasons as above, the complexity of the algorithm is increased significantly. We conclude that although the problem with the linear constraint is mathematically attractive, it is an extremely difficult computational problem. Unless an efficient algorithm can be developed for the indefinite systems of equations, this approach should be avoided.

5.2 Determining Both Parameters

In Section 5.1, we have demonstrated that the linearly constrained problem (5.2), is a difficult computational problem. One of the main reason was that $\lambda \in NBV(\Omega)$, implying that the discrete version of the Lagrange multiplier $\lambda \in \Re^N$, where N is on the order of $N_x N_y N_z$. Hence, there exists a scalar multiplier for every pixel in the image that must be computed. If we can devise a method so that λ is of smaller dimension, the resulting algorithm must be simpler then the one presented in Section 5.1.

A method to achieve this is to quadratically constrain the problem so that the constraining space is $\Re$ and therefore the resulting Lagrange multiplier $\lambda \in \Re^* = \Re$. If we have two quadratic constraints, the constraining space becomes $\Re^2$ and the multiplier $\lambda \in \Re^2$. We pose the quadratically constrained problem with both

incompressibility and divergence-free constraints below:

$$\begin{aligned} \textit{minimize} \quad & \int_\Omega (u_x^2 + u_y^2 + u_z^2 + v_x^2 + v_y^2 + v_z^2 + w_x^2 + w_y^2 + w_z^2)\, d\Omega \\ \textit{subject to} \quad & \int_\Omega (f_x u + f_y v + f_z w + f_t)^2\, d\Omega \leq \delta_1 \\ \textit{and} \quad & \int_\Omega (u_x + v_y + w_z)^2\, d\Omega \leq \delta_2 \end{aligned} \tag{5.5}$$

where (δ_1, δ_2) are the mean squared error terms that depend on the noise in the data f—i.e., δ_1—and the round-off noise during the computation—i.e., both δ_1 and δ_2. For the problem above, the resulting Lagrange multiplier $\lambda \in \Re^2$ since there are two quadratic constraints. Both constraints are included here, since by constraining the problem quadratically, having one or two constraints does not significantly alter the problem and the method of obtaining its solution. The solution to the above quadratically constrained problem is discussed in this section.

5.2.1 Constrained Optimization of Functionals by Dual Space Method

In this section, we establish the notation and summarize some relevant results from the optimization theory. For details, see [53, 41].

We discuss some results based on a general optimization problem below:

$$\begin{aligned} \textit{minimize} \quad & g(\xi) \\ \textit{subject to} \quad & G(\xi) \leq 0 \end{aligned} \tag{5.6}$$

where $g : D \rightarrow \Re, G : D \rightarrow Z$, and $D \subset X$, a vector space. And we assume that

there exists a closed positive cone $P \subset Z$, a constraining space, so that vector inequalities make sense.

Let $\xi_0 \in D \subset X$ solve (5.6) and $\eta_0^* \in Z^*$, the dual space of Z, be the Lagrange multiplier for ξ_0. Then, the solution (ξ_0, η_0^*) is characterized by

$$L(\xi_0, \eta^*) \leq L(\xi_0, \eta_0^*) \leq L(\xi, \eta_0^*), \qquad \eta_0^* \geq 0 \tag{5.7}$$

where L is the Lagrangian, for the minimization problem (5.6), as shown below:

$$L(\xi, \eta^*) = g(\xi) + \eta^* G(\xi).$$

In other words, the solution (ξ_0, η_0^*) is a saddle point of the Lagrangian L.

A generalized version of (5.7) is the Kuhn-Tucker condition, which briefly states that the Lagrangian

$$L(\xi, \eta_0^*) = g(\xi) + \eta_0^* G(\xi) \tag{5.8}$$

is stationary at ξ_0; and furthermore, $\eta_0^* G(\xi) = 0$. If g and G are Gateaux differentiable, from (5.8), we have $\delta g(\xi; h) + \eta_0^* \delta G(\xi; h) = 0$.

Suppose the constraining space $Z = \Re^n$. Then the Lagrange multiplier $\eta^* = \lambda = [\lambda_1, \ldots, \lambda_n]^T \in \Re^n$, since the dual of $\Re^n$ is $\Re^n$. In this case, the Lagrangian can be written as

$$L(\xi, \lambda) = g(\xi) + \lambda^T G(\xi).$$

and the Kuhn-Tucker condition becomes

$$\delta g(\xi; h) + \lambda^T \delta G(\xi; h) \;=\; 0 \tag{5.9}$$

$$\lambda^T G(\xi; h) = 0 \tag{5.10}$$

An important method in optimization theory is based on the dual functional. The primal functional ω and its dual functional ϕ are shown below:

$$\begin{aligned} \omega(\eta) &= \inf_{\xi \in D} \{g(\xi) : G(\xi) \leq \eta\} \\ \phi(\eta^*) &= \inf_{\xi \in D} \{g(\xi) + \eta^* G(\xi)\} \end{aligned} \tag{5.11}$$

It can be shown that ω is convex where as ϕ is concave.

The dual space method states that solving the problem in the primal space, e.g., by (5.9) and (5.10) above, is equivalent to first obtaining the optimal Lagrange multiplier by maximizing the dual functional ϕ, i.e.,

$$\max_{\lambda \geq 0} \phi(\lambda), \tag{5.12}$$

then solving (5.9) with the optimum λ.

The dual functional ϕ as written in (5.11) is not useful for our analysis. We find another approach for expressing ϕ by defining ξ_λ as the element such that

$$L(\xi_\lambda, \lambda) \leq L(\xi, \lambda), \qquad \forall \xi \in D \subset X \tag{5.13}$$

Since $\lambda \geq 0$, if g and G are convex, then for every fixed λ, there exists a minimizer ξ_λ of the Lagrangian $L(\xi, \lambda)$; and if g is strictly convex and G is convex, then this ξ_λ is unique.

As long as the element ξ_λ exists, the dual functional can be shown to be

$$\phi(\lambda) = g(\xi_\lambda) + \lambda^T G(\xi_\lambda) \tag{5.14}$$

This is the form we will be using in the next section to iteratively compute the Lagrange multiplier λ.

5.2.2 Constrained 3-D Motion Recovery Problem

We now have sufficient background to solve problem (5.5), repeated below including some notations of the previous section:

$$\begin{aligned} \textit{minimize} \quad & e_S(\mathbf{s}) = \int_\Omega (u_x^2 + u_y^2 + u_z^2 + v_x^2 + v_y^2 + v_z^2 \\ & \qquad + w_x^2 + w_y^2 + w_z^2)\, d\Omega \qquad (5.15) \\ \textit{subject to} \quad & G(\mathbf{s}) = \begin{bmatrix} \int_\Omega (f_x u + f_y v + f_z w + f_t)^2\, d\Omega \\ \int_\Omega (u_x + v_y + w_z)^2\, d\Omega \end{bmatrix} - \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix} \leq \mathbf{0} \end{aligned}$$

where $\mathbf{s} = (u, v, w)$, and (δ_1, δ_2) are the mean squared error terms.

From the Kuhn-Tucker condition, (5.9) and (5.10), we have,

$$\delta e_S(\mathbf{s};\mathbf{h}) + \lambda_1 \delta e_I(\mathbf{s};\mathbf{h}) + \lambda_2 \delta e_D(\mathbf{s};\mathbf{h}) = 0 \tag{5.16}$$

$$[\lambda_1, \lambda_2]\, G(\mathbf{s}) = 0 \tag{5.17}$$

As stated in the previous section, simultaneously satisfying the Kuhn-Tucker condition above is identical to first obtaining the optimal multiplier, $\lambda_0 = [\lambda_{1,0}, \lambda_{2,0}]^T$, then solving (5.16) with λ_0.

The optimal multiplier λ_0 can be obtained iteratively with the help of the dual

functional

$$\phi(\lambda) = g(\mathbf{s}_\lambda) + \lambda^T G(\mathbf{s}_\lambda) \tag{5.18}$$

where $\mathbf{s}_\lambda$ is the minimizer of the Lagrangian $L(\mathbf{s}, \lambda)$, for a fixed λ.

Since the dual functional ϕ is concave, the optimal λ_0 can be computed from

$$\phi'(\lambda) = 0$$

where $'$ denotes the Frechet derivative with respect to λ. But, using (5.18), we have [45, p. 398]:

$$\phi'(\lambda) \;=\; \left(g'(\mathbf{s}_\lambda) + \lambda^T G'(\mathbf{s}_\lambda)\right) \mathbf{s}'_\lambda + G(\mathbf{s}_\lambda)$$

The first term above vanishes by definition of $\mathbf{s}_\lambda$. Therefore, we obtain

$$\phi'(\lambda) = G(\mathbf{s}_\lambda)$$

and λ may be found from

$$G(\mathbf{s}_\lambda) = 0.$$

In order to search for the root of $G(\mathbf{s}_\lambda)$, we first need to evaluate $\mathbf{s}_\lambda$. The minimizer $\mathbf{s}_\lambda$, of the Lagrangian $L(\mathbf{s}, \lambda)$, for fixed λ, can be obtained by setting the Gateaux differential of the Lagrangian $L(\mathbf{s}, \lambda)$ to zero. We see that $\mathbf{s}_\lambda$ is exactly the solution to (5.16).

In Section 3.4 the solution to (5.16) with fixed $\lambda = [\lambda_1, \lambda_2]^T$ was provided where λ's were replaced by γ's. A numerical solution was given for this problem in Chapter 4. Therefore, to obtain $\mathbf{s}_\lambda$ we would simply use the conjugate gradient

algorithm as presented in Chapter 4.

The algorithm can be summarized as follows where we use $\mathbf{x}$ as the discrete version of $\mathbf{s}$.

- Choose an arbitrary $\lambda^{(0)} \neq 0$.
- Loop over $m = 0, 1, 2, \ldots$

    1. Obtain $\mathbf{x}_{\lambda^{(m)}}$, via some iterative algorithm.
    2. Update: $\lambda^{(m+1)} = \lambda^{(m)} + \mathbf{K}\, G(\mathbf{x}_{\lambda^{(m)}})$.
    3. Stop at a desired accuracy.

- The solution is the vector $\mathbf{x}_{\lambda^{(M)}}$ where M is the last m of the loop.

The step-size $\mathbf{K}$ can either be a constant matrix or some matrix function. In particular, for Newton's method,

$$\mathbf{K} = -\left[G'(\mathbf{x}_{\lambda^{(m)}})\right]^{-1}$$

where $'$ indicates the Frechet derivative. It can be shown that the inverse of the Frechet derivative of G is

$$\left[G'(\mathbf{x}_{\lambda^{(m)}})\right]^{-1} = \left(\frac{\partial G_1}{\partial \lambda_1}\frac{\partial G_2}{\partial \lambda_2} - \frac{\partial G_1}{\partial \lambda_2}\frac{\partial G_2}{\partial \lambda_1}\right)^{-1} \begin{bmatrix} \dfrac{\partial G_1}{\partial \lambda_1} & -\dfrac{\partial G_1}{\partial \lambda_2} \\ -\dfrac{\partial G_2}{\partial \lambda_2} & \dfrac{\partial G_1}{\partial \lambda_2} \end{bmatrix} \tag{5.19}$$

where $G = [G_1, G_2]^T$.

The discrete approximation of the partials above yields the secant method.

For instance, we approximate

$$\frac{\partial G_1}{\partial \lambda_1} = \frac{G_1^{(m)} - G_1^{(m-1)}}{\lambda_1^{(m)} - \lambda_1^{(m-1)}}$$

5.2.3 Results

In this section, we present the results of the 3-D motion recovery problem using optimization by the dual space method as presented in the previous section. To obtain $\mathbf{x}_\lambda$, we used the conjugate gradient algorithm of Chapter 4. The Lagrange multipliers were updated utilizing the secant method of the previous section.

This algorithm was tested on the images of Experiment 1 of Figure 4.1. The algorithm was run for four different scenarios and we present the results in Figures 5.1 through Figure 5.4. In all of these figures, (a) shows the first multiplier $\lambda_1^{(k)}$ as a function of the iteration number k; (b) shows the second multiplier $\lambda_2^{(k)}$ as a function of the iteration number k; (c) shows the progression of the log of the error in the incompressibility constraint, i.e.,

$$\int_\Omega (f_x u + f_y v + f_z w + f_t)^2 d\Omega;$$

and (d) shows the progression of the log of the error in the divergence-free constraint, i.e.,

$$\int_\Omega (u_x + v_y + w_z)^2 d\Omega.$$

The iteration number k indicates the loop index for the conjugate gradient algorithm. The multipliers were updated only when the residual $\| \mathbf{A}\mathbf{x}_{\lambda(k)} - \mathbf{b} \|$ was sufficiently small, i.e., when $\mathbf{x}_{\lambda(k)}$ is approximately equal to $\mathbf{A}^{-1}\mathbf{b}$.

The first set of experiments shown in Figures 5.1 and 5.2 demonstrates a case

where both constraints are active. Figure 5.1 shows the case where $(\delta_1, \delta_2) = (.05, 2.5)$ and the initial multiplier $\lambda^{(0)} = [.02, 60]^T$. It is seen that λ_1 and λ_2 converge in about 400 iterations. As expected, the error in the incompressibility constraint approaches δ_1 and the error in the divergence-free constraint approaches δ_2.

Figure 5.2 shows the case where (δ_1, δ_2) are the same as in Figure 5.1, but used a different initial multiplier $\lambda^{(0)} = [.003, 100]^T$. Again, the convergence of the multiplier is obtained in about 400 iterations. Since (δ_1, δ_2) were identical in both of these experiments, the multiplier converged to the same value as in Figure 5.1, as expected. Moreover, as in Figure 5.1, the errors in the respective constraints approach δ_1 and δ_2.

The next set of experiments shown in Figures 5.3 and 5.4 is a study of the active versus inactive constraints. Suppose that δ_2 is made 'large'. Then, we would expect the divergence-free constraint to become inactive. This would show up as $\lambda_2^{(k)} \rightarrow 0$. This is exactly what happens as evidenced by Figure 5.3. It shows that $\lambda_2^{(k)} \rightarrow 0$ as $k \rightarrow \infty$.

Figure 5.4 is the case where δ_1 is made 'large'. As expected, the incompressibility constraint becomes inactive. It is seen that $\lambda_1^{(k)} \rightarrow 0$ as $k \rightarrow \infty$.

From these examples, we infer that one must be judicious in choosing the mean-squared error terms δ_1 and δ_2.

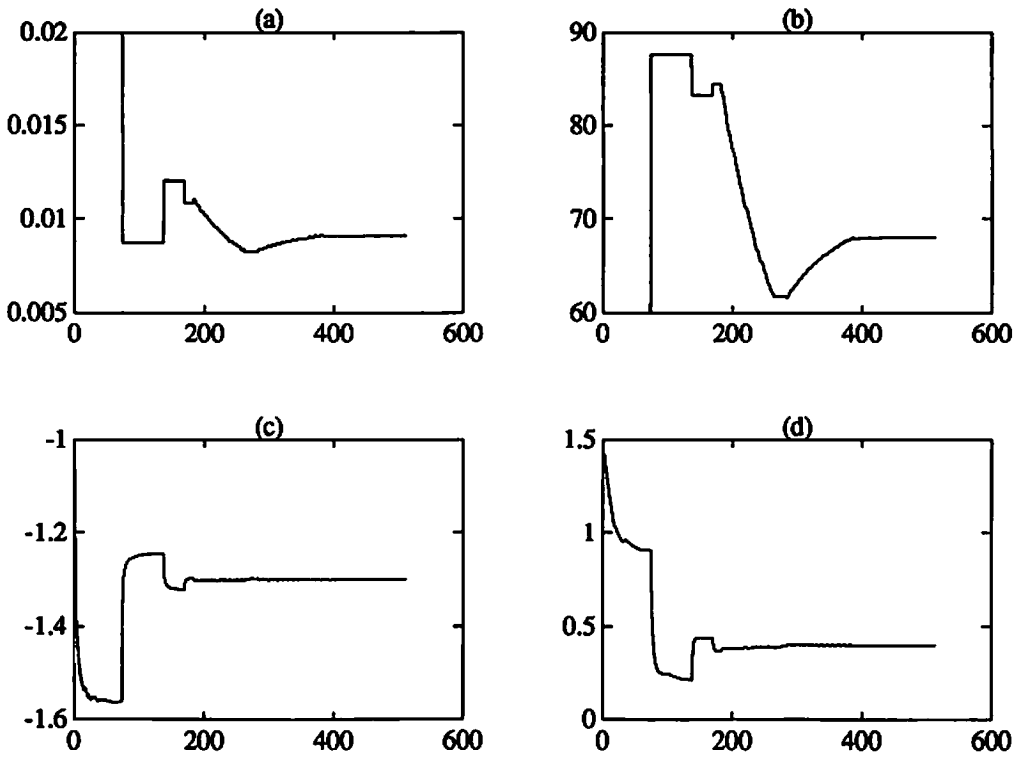


Figure 5.1: Results of the optimization by dual space method with $(\delta_1, \delta_2) = (.05, 2.5)$ and $(\lambda_1^{(0)}, \lambda_2^{(0)}) = (.02, 60)$. (a) $\lambda_1^{(k)}$ as a function of the iteration number k. (b) $\lambda_2^{(k)}$ as a function of k. (c) The log of the mean squared error in the incompressibility constraint as a function of k. (d) The log of the mean squared error in the divergence-free constraint as a function of k.

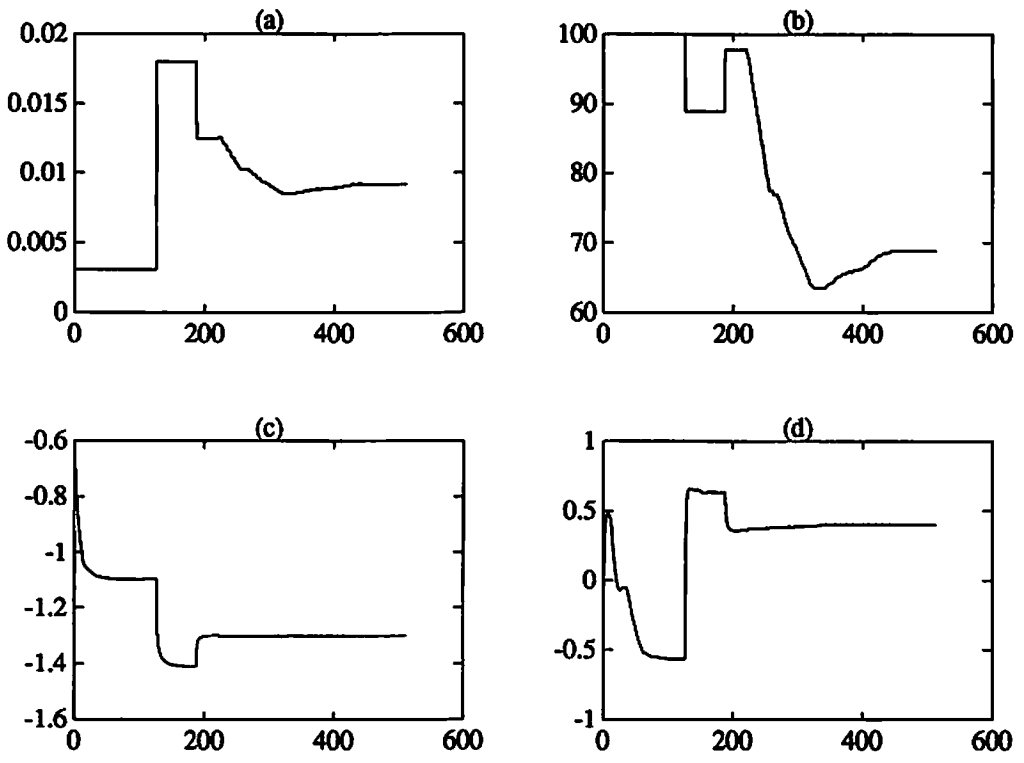


Figure 5.2: Results of the optimization by dual space method with $(\delta_1, \delta_2) = (.05, 2.5)$ and $(\lambda_1^{(0)}, \lambda_2^{(0)}) = (.003, 100)$. (a) $\lambda_1^{(k)}$ as a function of the iteration number k. (b) $\lambda_2^{(k)}$ as a function of k. (c) The log of the mean squared error in the incompressibility constraint as a function of k. (d) The log of the mean squared error in the divergence-free constraint as a function of k.

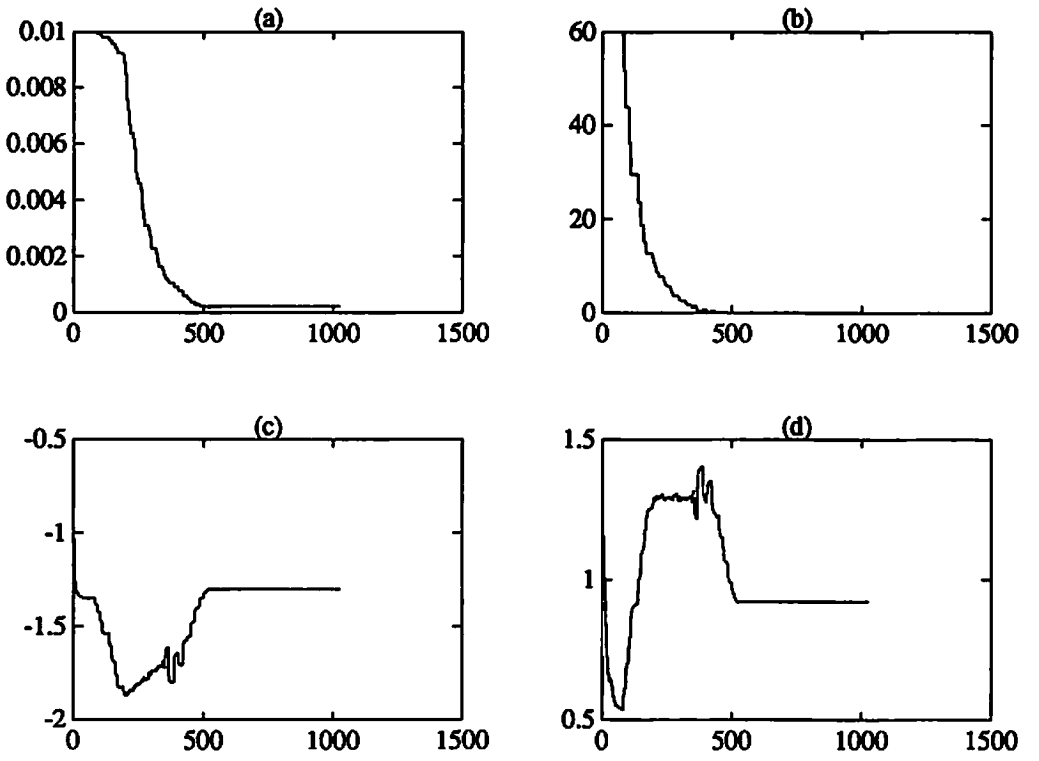


Figure 5.3: Results of the optimization by dual space method with $(\delta_1, \delta_2) = (.05, 20)$ and $(\lambda_1^{(0)}, \lambda_2^{(0)}) = (.01, 60)$. (a) $\lambda_1^{(k)}$ as a function of the iteration number k. (b) $\lambda_2^{(k)}$ as a function of k. (c) The log of the mean squared error in the incompressibility constraint as a function of k. (d) The log of the mean squared error in the divergence-free constraint as a function of k.

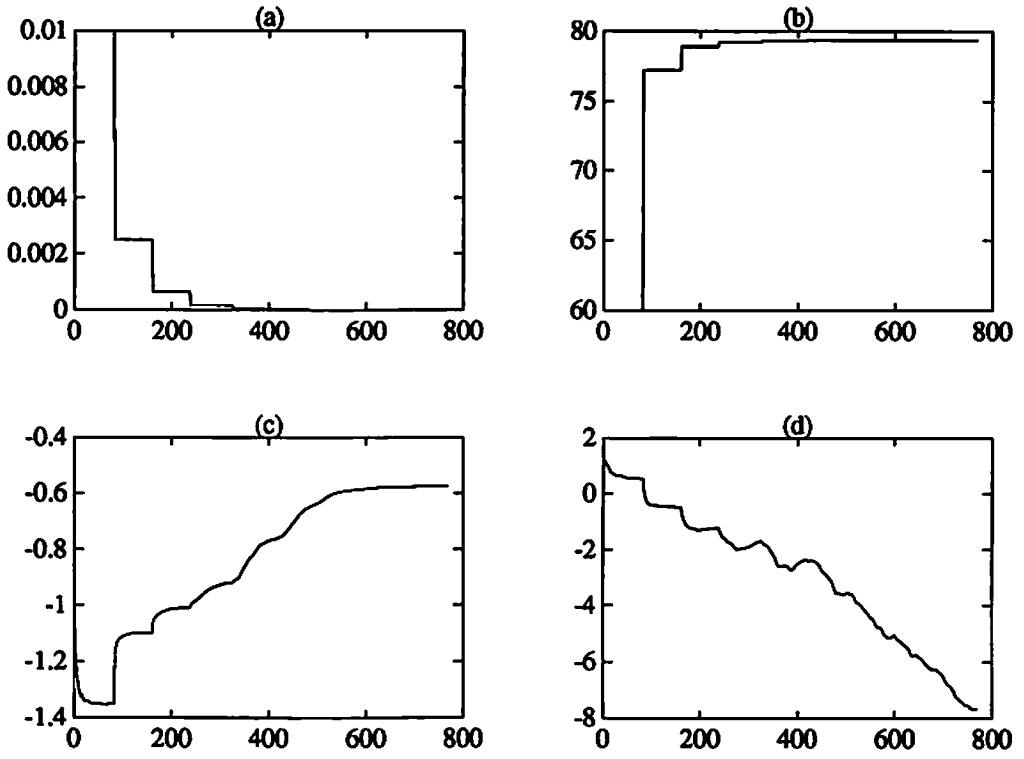


Figure 5.4: Results of the optimization by dual space method with $(\delta_1, \delta_2) = (.5, .00001)$ and $(\lambda_1^{(0)}, \lambda_2^{(0)}) = (.01, 60)$. (a) $\lambda_1^{(k)}$ as a function of the iteration number k. (b) $\lambda_2^{(k)}$ as a function of k. (c) The log of the mean squared error in the incompressibility constraint as a function of k. (d) The log of the mean squared error in the divergence-free constraint as a function of k.

5.3 Discussion

In this chapter, we have posed the 3-D motion recovery problem as a constrained optimization problem. We have considered both the linearly constrained problem and the quadratically constrained problem. It was demonstrated that the linearly constrained problem, although mathematically simpler, results in a difficult computational problem.

The quadratically constrained problem was solved via the dual-space method and some results were shown in Section 5.2.3. These results indicated the importance of the mean-squared error terms δ_1 and δ_2. In Figures 5.3 and 5.4 we showed that δ_1 and/or δ_2 cannot be made too large as the corresponding constraint becomes inactive. On the other hand, δ_1 and δ_2 cannot be made too small as this will tend to drive the multipliers to diverge.

Therefore, the mean-squared error terms (δ_1, δ_2) must be chosen so that the constraints remain active—i.e., small enough (δ_1, δ_2)—and so that the problem is not overly constrained—i.e., large enough (δ_1, δ_2). This difficult choice of (δ_1, δ_2) remains an important theoretical question. It seems possible that (δ_1, δ_2) may be obtained by some means of statistical analysis based on the image formation model.

We have thus converted the problem of obtaining the parameters γ_1 and γ_2 of Chapters 3 and 4 into obtaining the mean-squared error terms δ_1 and δ_2. At this point, the value of this conversion remains unclear. However, making an educated guess at the mean-squared error terms (which we have some physical feel for) seem more appealing than the widely practiced approach of choosing the parameters γ_1 and γ_2 simply as regularization or smoothing parameters.

Chapter 6

Conclusions and Future Research

6.1 Conclusions

The goal of this work was to quantify the motion of a beating heart given a cine CT sequence of cardiac images. Since all motion is completely characterized by the velocity field generated by the motion, knowing (or computing) the velocity field completely quantifies the motion. This work was a first attempt at computing the velocity field generated by a cine CT sequence and thus quantifying the motion of a beating heart. However, the algorithm developed also works for other types of images as long as the images are density images of incompressible objects. The algorithm presented in this dissertation should result in a solution that is agreeable when applied to density images.

While we have shown that reasonable velocity fields can be computed from a sequence of 3-D images, we make no claim as to the correctness of the computed field. In fact, if the inner circle of the experiment 1 had been rotating about its center, both images (frames 1 and 2) will be identical and naturally, trying to

compute the rotational motion would be impossible. Even the human observer of this situation would perceive a zero motion. This is where the concept of *apparent motion* [58] needs to be addressed. What we have computed is the apparent velocity field and not the actual velocity field. Although these two fields can be quite different in situations such as above, we do have to depend on the fact that in most situations, the two fields are not too different from each other [58].

For a certain class of images and its underlying motion, it may be possible to incorporate other *a priori* knowledge into the formulation so that the problem is well-posed, without the smoothness assumption This approach will be motion and/or image dependent and will not result in a general algorithm such as the one presented in this dissertation. However, if such an algorithm is developed, then we would be computing the true velocity field rather than the apparent velocity field.

It remains to be seen whether the method in this dissertation will be useful in a clinical setting helping to assess the motion of the heart, either in detecting certain abnormalities or in obtaining a better understanding of the heart function. Figures 4.10 and 4.11 are only visualizations of the solution. Each overlayed vector represents three numbers (three spatial components). It is hoped that, in conjunction with a suitable segmentation algorithm, various heart parameters such as the heart pump rate and the flux across the heart valves can reliably be determined from the computed velocity field.

6.2 Future Research

Most research problems are never completely solved; and this dissertation is not an exception. It is hoped that this dissertation will spawn some studies extending this work. There are many facets of this work that warrant further study.

First, the development of a fast algorithm is crucial for the clinical use of this procedure. In Chapter 4, a preconditioner was used and shown to converge faster; however, we have not provided a mathematical proof explaining this observation as the system matrix is difficult to analyze. Since the system matrix includes the image dependent terms, it is possible that the preconditioner was effective for the images tested thus far. Perhaps, we should find other avenues in speeding up the algorithm. As preconditioners are known to converge faster for higher frequency components of the solution [47], the ultimate approach may be to develop a multi-grid algorithm in conjunction with a preconditioner to speed up the convergence. Iterating between the coarse and fine grid should allow the lower frequency components to converge faster.

Second, a method of determining the mean-squared errors δ_1 and δ_2 is needed. As defined by Tikhonov [3], the regularization parameter is a positive weight assigned to the stabilizing functional. In our problem, there are two parameters that affect the weighting of the stabilizing functional. Hence, these parameters can be considered as regularization parameters. The difference, if any, is in the semantics; and therefore, there was no attempt in differentiating the two in this dissertation. The method developed in Chapter 5 is the first application of the dual-space method in obtaining the so-called regularization parameters. Although

the dual-space method proved to be useful in updating these parameters, *a priori* knowledge of the mean-squared errors was required. We need an approach to estimate these mean-squared errors. A statistical approach, based on some image formation model, may perhaps be used to determined the mean-squared errors.

And finally, a quantitative verification (or even determining the magnitude of deviation from the true velocity field) of the algorithm would definitely be of value. While this dissertation has shown that reasonable looking fields can be obtained from density images, we have not provided any quantitative comparisons. In order to quantitatively evaluate the algorithm, we need to consider a system where the velocity field is known *a priori*. This may be done by considering a simple fluid dynamical system where a closed-form solution to the Navier-Stokes equation is available. The computed velocity field can then be compared to the actual closed form solution velocity field. Examples include the flow between parallel plates and the Couette flow [15]. The numerical solution based on the images of these systems can be compared to the closed-form solution for evaluation. This comparative study should be conducted in conjunction with determining the regularization parameters (γ_1, γ_2) or the mean-squared error terms (δ_1, δ_2). Once the computed velocity field is determined to be a reasonable approximation to the true velocity field, local cardiac deformation properties as well as the pressure distribution within the heart chambers may be computed by using the Navier-Stokes equation.

Appendix A

Euler-Lagrange Equations for Certain Optimization Problems

A.1 Unconstrained Problems

We present a derivation of the Euler-Lagrange equations for unconstrained optimization problems.

We derive the Euler-Lagrange equations for optimizing the functional

$$e(\mathbf{s}) = \int_{\Omega} F(\mathbf{s}, \partial\mathbf{s})\, d\Omega$$

where

$$\begin{aligned} \mathbf{s} &= (u, v, w) \\ \partial\mathbf{s} &= (u_x, u_y, u_z, v_x, v_y, v_z, w_x, w_y, w_z) \\ \Omega &= [0, T_x] \times [0, T_y] \times [0, T_z] \end{aligned}$$

$$d\Omega \;=\; dx\,dy\,dz$$

Assuming that e is a Gateaux differentiable functional on a vector space X, a necessary condition [53] for e to achieve an extremum at $\mathbf{s}_{ext} \in X$ is that the Gateaux differential

$$\delta e(\mathbf{s}_{ext};\mathbf{h}) = 0 \qquad\qquad \forall \mathbf{h} \in X$$

The problem is set in the space of continuously differentiable vector fields, $\mathbf{C}^1(\Omega)$, to ensure the existence of the penalty functional $e(\mathbf{s},\partial\mathbf{s})$, i.e., $\mathbf{s}$, $\mathbf{h}=(h_1,h_2,h_3) \in \mathbf{C}^1(\Omega)$. By the chain rule of partial differentiation,

$$\begin{aligned}\frac{d}{d\alpha}F(\mathbf{s}+\alpha\mathbf{h},\partial(\mathbf{s}+\alpha\mathbf{h}))\Big|_{\alpha=0} \;=\; & F_u h_1 + F_{u_x} h_{1x} + F_{u_y} h_{1y} + F_{u_z} h_{1z} \\ & + F_v h_2 + F_{v_x} h_{2x} + F_{v_y} h_{2y} + F_{v_z} h_{2z} \\ & + F_w h_3 + F_{w_x} h_{3x} + F_{w_y} h_{3y} + F_{w_z} h_{3z}\end{aligned}$$

Therefore,

$$\begin{aligned}\delta e(\mathbf{s};\mathbf{h}) \;&=\; \int_\Omega \frac{d}{d\alpha}F(\mathbf{s}+\alpha\mathbf{h},\partial(\mathbf{s}+\alpha\mathbf{h}))\,d\Omega \;\Big|_{\alpha=0} \\ &=\; \int_\Omega (F_u h_1 + F_{u_x} h_{1x} + F_{u_y} h_{1y} + F_{u_z} h_{1z})\,d\Omega \\ &\quad + \int_\Omega (F_v h_2 + F_{v_x} h_{2x} + F_{v_y} h_{2y} + F_{v_z} h_{2z})\,d\Omega \\ &\quad + \int_\Omega (F_w h_3 + F_{w_x} h_{3x} + F_{w_y} h_{3y} + F_{w_z} h_{3z})\,d\Omega\end{aligned}$$

All three integrals above may be simplified using the divergence theorem of Gauss.

For instance, we apply the divergence theorem to the first integral.

$$\int_\Omega (F_u h_1 + F_{u_x} h_{1x} + F_{u_y} h_{1y} + F_{u_z} h_{1z})\, d\Omega$$

$$= \int_\Omega \{h_1(F_u - \frac{\partial}{\partial x}F_{u_x} - \frac{\partial}{\partial y}F_{u_y} - \frac{\partial}{\partial z}F_{u_z}) \quad + \quad \nabla \cdot \begin{bmatrix} h_1 F_{u_x} \\ h_1 F_{u_y} \\ h_1 F_{u_z} \end{bmatrix} \} \, d\Omega$$

$$= \int_\Omega h_1(F_u - \frac{\partial}{\partial x}F_{u_x} - \frac{\partial}{\partial y}F_{u_y} - \frac{\partial}{\partial z}F_{u_z})\, d\Omega \quad + \quad \oint_{\partial\Omega} h_1 \begin{bmatrix} F_{u_x} \\ F_{u_y} \\ F_{u_z} \end{bmatrix} \cdot d\mathbf{n}$$

where

$$\begin{aligned} \partial\Omega &= \text{surface enclosing } \Omega \\ d\mathbf{n} &= \text{surface normal differential} \end{aligned}$$

Similar application of the divergence theorem to the other integrals yield

$$\begin{aligned} \delta e(\mathbf{s};\mathbf{h}) = &\int_\Omega h_1(F_u - \frac{\partial}{\partial x}F_{u_x} - \frac{\partial}{\partial y}F_{u_y} - \frac{\partial}{\partial z}F_{u_z})\, d\Omega \\ &+ \int_\Omega h_2(F_v - \frac{\partial}{\partial x}F_{v_x} - \frac{\partial}{\partial y}F_{v_y} - \frac{\partial}{\partial z}F_{v_z})\, d\Omega \\ &+ \int_\Omega h_3(F_w - \frac{\partial}{\partial x}F_{w_x} - \frac{\partial}{\partial y}F_{w_y} - \frac{\partial}{\partial z}F_{w_z})\, d\Omega \\ &+ \oint_{\partial\Omega} \left(h_1 \begin{bmatrix} F_{u_x} \\ F_{u_y} \\ F_{u_z} \end{bmatrix} + h_2 \begin{bmatrix} F_{v_x} \\ F_{v_y} \\ F_{v_z} \end{bmatrix} + h_3 \begin{bmatrix} F_{w_x} \\ F_{w_y} \\ F_{w_z} \end{bmatrix} \right) \cdot d\mathbf{n} \end{aligned} \tag{A.1}$$

It is typical that the solution minimizing the functional $e(\mathbf{s})$ is often a system of

PDEs. Then, in order to completely specify the system, a boundary condition must be provided.

Dirichlet boundary condition

Often, the values of the extremum $\mathbf{s}_{ext}$ on $\partial\Omega$ is known *a priori*. This boundary condition is called the *essential boundary condition* [59] or the Dirichlet boundary condition (actually, these two terms are used interchangeably). In this case, all $\mathbf{h} = (h_1, h_2, h_3) \in \mathbf{C}^1(\Omega)$ in the set of admissible $\mathbf{h}$ must vanish on $\partial\Omega$. Therefore, the flux integral is identical to zero.

Moreover, since the Gateaux differential $\delta e(\mathbf{s}; \mathbf{h})$ must be zero at the extremum $\mathbf{s}_{ext}$ for any $\mathbf{h}$ in the admissible set, it follows that in the first three integrands, the terms inside the parenthesis of the volume integral must be zero and the Euler-Lagrange equations are provided below:

$$\begin{aligned} 0 &= F_u - \frac{\partial}{\partial x}F_{u_x} - \frac{\partial}{\partial y}F_{u_y} - \frac{\partial}{\partial z}F_{u_z} \\ 0 &= F_v - \frac{\partial}{\partial x}F_{v_x} - \frac{\partial}{\partial y}F_{v_y} - \frac{\partial}{\partial z}F_{v_z} \\ 0 &= F_w - \frac{\partial}{\partial x}F_{w_x} - \frac{\partial}{\partial y}F_{w_y} - \frac{\partial}{\partial z}F_{w_z} \end{aligned} \tag{A.2}$$

with a Dirichlet boundary condition.

Natural boundary condition

If no *a priori* knowledge is available, we have no *a priori* boundary conditions. In this case, we can show that the solution must satisfy the *natural boundary condition*. We can obtain the Euler-Lagrange equations with this boundary condition

by generalizing the method in [59, pp. 73–74].

Suppose there exists some $\mathbf{s}_{ext}$ such that the Gateaux differential $\delta e(\mathbf{s};\mathbf{h})$ is zero (see Section 3.3 for existence and uniqueness of $\mathbf{s}_{ext}$). It must be zero for all $\mathbf{h} \in \mathbf{C}^1(\Omega)$, and in particular, it must also be zero for all $\mathbf{h}$ that vanish on $\partial\Omega$. But, for this particular case, the flux integral is zero (as discussed previously). Hence, the same Euler-Lagrange equations (A.2) must hold on the interior of Ω.

It follows that the volume integrals in (A.1) must be zero, and therefore, the flux integral must be zero for all $\mathbf{h} \in \mathbf{C}^1(\Omega)$. Moreover, since $\mathbf{h}$ is permitted to be any element in $\mathbf{C}^1(\Omega)$, there are no restrictions on the values of $\mathbf{h}$ on $\partial\Omega$. This gives us the natural boundary condition for this problem.

In summary, for the natural boundary condition (no boundary condition) case, we have the Euler-Lagrange equations (A.2) with the following natural boundary condition.

$$\begin{bmatrix} F_{u_x} \\ F_{u_y} \\ F_{u_z} \end{bmatrix} \cdot d\mathbf{n} = \begin{bmatrix} F_{v_x} \\ F_{v_y} \\ F_{v_z} \end{bmatrix} \cdot d\mathbf{n} = \begin{bmatrix} F_{w_x} \\ F_{w_y} \\ F_{w_z} \end{bmatrix} \cdot d\mathbf{n} = 0, \quad \text{on } \partial\Omega \qquad \text{(A.3)}$$

A.2 Constrained Problems

We present a derivation of the Euler-Lagrange equations for constrained optimization problems such as below:

$$\begin{aligned} &\textit{minimize} && J(\mathbf{s}) = \int_\Omega F(\mathbf{s}, \partial\mathbf{s}) d\Omega \\ &\textit{subject to} && \psi(u_x, v_y, w_z) = 0 \end{aligned}$$

where

$$\begin{aligned}
\mathbf{s} &= (u, v, w) \\
\partial\mathbf{s} &= (u_x, u_y, u_z, v_x, v_y, v_z, w_x, w_y, w_z) \\
\Omega &= [0, T_x] \times [0, T_y] \times [0, T_z] \\
d\Omega &= dx\, dy\, dz
\end{aligned}$$

The Gateaux differential

$$\begin{aligned}
\delta J(\mathbf{s}; \mathbf{h}) &= \int_\Omega \frac{d}{d\alpha} F(\mathbf{s} + \alpha\mathbf{h}, \partial(\mathbf{s} + \alpha\mathbf{h}))\, d\Omega \Big|_{\alpha=0} \\
&= \int_\Omega (F_u h_1 + F_{u_x} h_{1x} + F_{u_y} h_{1y} + F_{u_z} h_{1z})\, d\Omega \\
&\quad + \int_\Omega (F_v h_2 + F_{v_x} h_{2x} + F_{v_y} h_{2y} + F_{v_z} h_{2z})\, d\Omega \\
&\quad + \int_\Omega (F_w h_3 + F_{w_x} h_{3x} + F_{w_y} h_{3y} + F_{w_z} h_{3z})\, d\Omega
\end{aligned}$$

As in Appendix A.1, the problem is set in the space of continuously differentiable vector fields, $\mathbf{C}^1(\Omega)$, to ensure the existence of the functional $J(\mathbf{s}, \partial\mathbf{s})$, i.e., $\mathbf{s}$, $\mathbf{h} = (h_1, h_2, h_3) \in \mathbf{C}^1(\Omega)$.

Consider the constraining functional ψ as a mapping $H : \mathbf{C}^1(\Omega) \mapsto Z = \mathbf{C}^1(\Omega)$. Then, the Gateaux differential

$$\begin{aligned}
\delta H(\mathbf{s}; \mathbf{h}) &= \psi_{u_x} h_{1x} + \psi_{v_y} h_{2y} + \psi_{w_z} h_{3z} \\
&= h_{1x} + h_{2y} + h_{3z}
\end{aligned}$$

where we have assumed that $\psi(\mathbf{s}) = \nabla \cdot \mathbf{s} = u_x + v_y + w_z$.

From the Lagrange Multiplier Theorem [53, p. 243], there exists

$$\eta_0^* \in Z^* = (\mathbf{C}^1(\Omega))^* = NBV(\Omega)$$

such that the Lagrangian

$$L(\mathbf{s}) = J(\mathbf{s}) + \eta_0^* H(\mathbf{s})$$

is stationary at a local extremum $\mathbf{s}_{ext}$, i.e.,

$$\delta L(\mathbf{s}_{ext}; \mathbf{h}) = \delta J(\mathbf{s}_{ext}; \mathbf{h}) + \eta_0^* \delta H(\mathbf{s}_{ext}; \mathbf{h}) = 0$$

If the Lagrangian L is convex, then $\mathbf{s}_{ext}$ is a global minimum. The Lagrange multiplier η_0^* is an element of Z^*, the dual space of the constraining space Z. It is described by

$$\eta_0^* p = \int_\Omega \lambda(x,y,z)\, p(x,y,z) d\Omega,$$

and it follows that

$$\eta_0^* \delta H(\mathbf{s}; \mathbf{h}) = \int_\Omega \lambda(x,y,z)\, (h_{1x} + h_{2y} + h_{3z}) d\Omega.$$

Therefore, the Gateaux differential of the Lagrangian is as follows:

$$\begin{aligned} \delta L(\mathbf{s}; \mathbf{h}) \;=\; & \int_\Omega (F_u h_1 + (F_{u_x} + \lambda)\, h_{1x} + F_{u_y} h_{1y} + F_{u_z} h_{1z})\, d\Omega \\ & + \int_\Omega (F_v h_2 + F_{v_x} h_{2x} + (F_{v_y} + \lambda)\, h_{2y} + F_{v_z} h_{2z})\, d\Omega \\ & + \int_\Omega (F_w h_3 + F_{w_x} h_{3x} + F_{w_y} h_{3y} + (F_{w_z} + \lambda)\, h_{3z})\, d\Omega \end{aligned}$$

As done previously in Appendix A.1, the above may be simplified using the divergence theorem of Gauss. This yields

$$\begin{aligned}\delta L(\mathbf{s};\mathbf{h}) = &\int_\Omega h_1(F_u - \frac{\partial}{\partial x}(F_{u_x}+\lambda) - \frac{\partial}{\partial y}F_{u_y} - \frac{\partial}{\partial z}F_{u_z})\,d\Omega \\ &+ \int_\Omega h_2(F_v - \frac{\partial}{\partial x}F_{v_x} - \frac{\partial}{\partial y}(F_{v_y}+\lambda) - \frac{\partial}{\partial z}F_{v_z})\,d\Omega \\ &+ \int_\Omega h_3(F_w - \frac{\partial}{\partial x}F_{w_x} - \frac{\partial}{\partial y}F_{w_y} - \frac{\partial}{\partial z}(F_{w_z}+\lambda))\,d\Omega \\ &+ \oint_{\partial\Omega}(h_1\begin{bmatrix} F_{u_x}+\lambda \\ F_{u_y} \\ F_{u_z}\end{bmatrix} + h_2\begin{bmatrix} F_{v_x} \\ F_{v_y}+\lambda \\ F_{v_z}\end{bmatrix} + h_3\begin{bmatrix} F_{w_x} \\ F_{w_y} \\ F_{w_z}+\lambda\end{bmatrix})\cdot d\mathbf{n}\end{aligned} \tag{A.4}$$

Using similar arguments as in Appendix A.1, the following Euler-Lagrange Equations with an appropriate boundary condition can be obtained:

$$\begin{aligned} 0 &= F_u - \frac{\partial}{\partial x}(F_{u_x}+\lambda) - \frac{\partial}{\partial y}F_{u_y} - \frac{\partial}{\partial z}F_{u_z} \\ 0 &= F_v - \frac{\partial}{\partial x}F_{v_x} - \frac{\partial}{\partial y}(F_{v_y}+\lambda) - \frac{\partial}{\partial z}F_{v_z} \\ 0 &= F_w - \frac{\partial}{\partial x}F_{w_x} - \frac{\partial}{\partial y}F_{w_y} - \frac{\partial}{\partial z}(F_{w_z}+\lambda)\end{aligned} \tag{A.5}$$

Again, as in Appendix A.1, if the values of $\mathbf{s}_{ext}$ is known on the boundary $\partial\Omega$, then we use the Dirichlet boundary condition.

If we have no knowledge on the values of $\mathbf{s}_{ext}$ on $\partial\Omega$, then we use the natural boundary condition given below:

$$\begin{bmatrix} F_{u_x}+\lambda \\ F_{u_y} \\ F_{u_z}\end{bmatrix}\cdot d\mathbf{n} = \begin{bmatrix} F_{v_x} \\ F_{v_y}+\lambda \\ F_{v_z}\end{bmatrix}\cdot d\mathbf{n} = \begin{bmatrix} F_{w_x} \\ F_{w_y} \\ F_{w_z}+\lambda\end{bmatrix}\cdot d\mathbf{n} = 0, \quad \text{on } \partial\Omega$$

Appendix B

Miscellaneous Proofs

In this appendix, we present the proofs for Lemma 3.5 and Lemma 3.6.

Proof of Lemma 3.5

Let g be convex. Then, by definition 3.3, for $\beta \in (0,1)$

$$g(\beta\eta + (1-\beta)\xi) \leq \beta g(\eta) + (1-\beta)g(\xi) \qquad \forall \xi, \eta \in X$$

Since $\beta \neq 0$,

$$\frac{g(\beta\eta + (1-\beta)\xi) - g(\xi)}{\beta} \leq g(\eta) - g(\xi)$$

It follows that

$$\frac{g(\xi - \beta(\eta - \xi)) - g(\xi)}{\beta} \leq g(\eta) - g(\xi)$$

By taking the limit $\beta \to 0$, the left hand side becomes the Gateaux differential $\delta g(\xi; \eta - \xi)$. Hence, we get the desired result

$$g(\eta) \geq g(\xi) + \delta g(\xi; \eta - \xi) \qquad \forall \xi, \eta \in X \tag{B.1}$$

Now, for converse, let (B.1) be true. With appropriate substitution of variables, it follows that

$$\begin{aligned} g(\xi_1) &\geq g(\xi) + \delta g(\xi; \xi_1 - \xi) \qquad \forall \xi_1, \xi \in X \\ g(\xi_2) &\geq g(\xi) + \delta g(\xi; \xi_2 - \xi) \qquad \forall \xi_2, \xi \in X \end{aligned}$$

Combining above two equations, it is easy to show that

$$\beta g(\xi_1) + (1 - \beta) g(\xi_2) \geq g(\xi) + \delta g(\xi; \beta \xi_1 + (1 - \beta)\xi_2 - \xi)$$

Now simply letting $\xi = \beta \xi_1 + (1 - \beta)\xi_2$ establishes the following desired result

$$\beta g(\xi_1) + (1 - \beta) g(\xi_2) \geq g(\beta \xi_1 + (1 - \beta)\xi_2) \qquad \forall \xi_1, \xi_2 \in X$$

The proof also holds if all inequalities are strict and the the elements of X were chosen such that $\xi \neq \eta, \xi \neq \xi_2$ and $\xi_1 \neq \xi_2$. Therefore, it follows that if the inequality in (3.14) is strict, then g is strictly convex. This completes the proof.

Proof of Lemma 3.6

The proof is presented in three parts, one part for each functional.

part i: $e_S(\mathbf{s})$ is convex.

Proof:

Making use of (3.11), the Gateaux differential of e_S can be computed as

$$\begin{aligned}
\delta e_S(\mathbf{s};\mathbf{h}) &= \frac{d}{d\alpha}\int_\Omega \{[(u+\alpha h_1)_x]^2 + [(u+\alpha h_1)_y]^2 + [(u+\alpha h_1)_z]^2 \\
&\qquad + [(v+\alpha h_2)_x]^2 + [(v+\alpha h_2)_y]^2 + [(v+\alpha h_2)_z]^2 \\
&\qquad + [(w+\alpha h_3)_x]^2 + [(w+\alpha h_3)_y]^2 + [(w+\alpha h_3)_z]^2\}\, d\Omega \;|_{\alpha=0} \\
&= \int_\Omega \frac{d}{d\alpha}\{[(u_x+\alpha h_{1x})]^2 + \cdots + [(w_z+\alpha h_{3z})]^2\}\, d\Omega \;|_{\alpha=0} \\
&= 2\int_\Omega \{(u_x+\alpha h_{1x})h_{1x} + \cdots + (w_z+\alpha h_{3z})h_{3z}\}\, d\Omega \;|_{\alpha=0} \\
&= 2\int_\Omega \{(u_x h_{1x} + u_y h_{1y} + u_z h_{1z}) + (v_x h_{2x} + v_y h_{2y} + v_z h_{2z}) \\
&\qquad + (w_x h_{3x} + w_y h_{3y} + w_z h_{3z})\}\, d\Omega \qquad \text{(B.2)}
\end{aligned}$$

where $\mathbf{s} = (u, v, w)$, and $\mathbf{h} = (h_1, h_2, h_3)$.

For every $\mathbf{s}_1 = (u_1, v_1, w_1)^T$ and $\mathbf{s}_2 = (u_2, v_2, w_2)^T$, it is obvious that

$$\begin{aligned}
0 \;\geq\; \int_\Omega \{&(u_{2x}-u_{1x})^2 + (u_{2y}-u_{1y})^2 + (u_{2z}-u_{1z})^2 \\
&+ (v_{2x}-v_{1x})^2 + (v_{2y}-v_{1y})^2 + (v_{2z}-v_{1z})^2 \\
&+ (w_{2x}-w_{1x})^2 + (w_{2y}-w_{1y})^2 + (w_{2z}-w_{1z})^2\}\, d\Omega
\end{aligned}$$

$$= \int_\Omega \{(u_{2x}^2 - 2u_{1x}u_{2x} + 2u_{1x}^2) + (u_{2y}^2 - 2u_{1y}u_{2y} + 2u_{1y}^2)$$

$$+ (u_{2z}^2 - 2u_{1z}u_{2z} + 2u_{1z}^2) + (v_{2x}^2 - 2v_{1x}v_{2x} + 2v_{1x}^2)$$

$$+ (v_{2y}^2 - 2v_{1y}v_{2y} + 2v_{1y}^2) + (v_{2z}^2 - 2v_{1z}v_{2z} + 2v_{1z}^2)$$

$$+ (w_{2x}^2 - 2w_{1x}w_{2x} + 2w_{1x}^2) + (w_{2y}^2 - 2w_{1y}w_{2y} + 2w_{1y}^2)$$

$$+ (w_{2z}^2 - 2w_{1z}w_{2z} + 2w_{1z}^2)\}\, d\Omega \qquad \text{(B.3)}$$

Rearrangement of terms in (B.3) yield

$$\int_\Omega (u_{2x}^2 + u_{2y}^2 + u_{2z}^2 + v_{2x}^2 + v_{2y}^2 + v_{2z}^2 + w_{2x}^2 + w_{2y}^2 + w_{2z}^2)\, d\Omega \qquad \text{(B.4)}$$

$$\geq \int_\Omega (u_{1x}^2 + u_{1y}^2 + u_{1z}^2 + v_{1x}^2 + v_{1y}^2 + v_{1z}^2 + w_{1x}^2 + w_{1y}^2 + w_{1z}^2)\, d\Omega \qquad \text{(B.5)}$$

$$+ 2\int_\Omega \{u_{1x}(u_{2x} - u_{1x}) + \cdots + w_{1z}(w_{2z} - w_{1z})\}\, d\Omega$$

The left hand side of (B.4) is $e_S(\mathbf{s}_2)$; and in view of (B.2), the right hand side works out to be $e_S(\mathbf{s}_1) + \delta e_S(\mathbf{s}_1; \mathbf{s}_2 - \mathbf{s}_1)$. Therefore, by Lemma 3.5, e_S is convex on $\mathbf{C}^1(\Omega)$.

part ii: $e_I(\mathbf{s})$ is convex.

Proof:

The Gateaux differential for e_I can be shown to be

$$\delta e_I(\mathbf{s}; \mathbf{h}) = 2\int_\Omega \{(f_x u + f_y v + f_z w)(f_x h_1 + f_y h_2 + f_z h_3)\}\, d\Omega \qquad \text{(B.6)}$$

Moreover, for every different $\mathbf{s}_1 = (u_1, v_1, w_1)$ and $\mathbf{s}_2 = (u_2, v_2, w_2)$,

$$\int_\Omega \{f_x(u_2 - u_1) + f_y(v_2 - v_1) + f_z(w_2 - w_1)\}^2 d\Omega \geq 0 \tag{B.7}$$

Some manipulation of terms show that (B.7) is equivalent to following.

$$\begin{aligned} &\int_\Omega (f_x u_2 + f_y v_2 + f_z w_2 + f_t)^2 d\Omega \\ &\geq \int_\Omega (f_x u_1 + f_y v_1 + f_z w_1 + f_t)^2 d\Omega \\ &\qquad + 2\int_\Omega \{(f_x u_1 + f_y v_1 + f_z w_1 + f_t) \\ &\qquad\qquad [f_x(u_2 - u_1) + f_y(v_2 - v_1) + f_z(w_2 - w_1)]\}\, d\Omega \end{aligned} \tag{B.8}$$

The left hand side of (B.8) is $e_I(\mathbf{s}_2)$; and in view of (B.6), the right hand side works out to be $e_I(\mathbf{s}_1) + \delta e_I(\mathbf{s}_1; \mathbf{s}_2 - \mathbf{s}_1)$. Therefore, by Lemma 3.5, e_M is convex on $\mathbf{C}^1(\Omega)$.

part iii: $e_D(\mathbf{s})$ is convex

Proof:

The Gateaux differential for e_D can be shown to be

$$\delta e_D(\mathbf{s}; \mathbf{h}) = 2\int_\Omega \{(u_x + v_y + w_z)(h_{1x} + h_{2y} + h_{3z})\}\, d\Omega \tag{B.9}$$

Moreover, for every different $\mathbf{s}_1 = (u_1, v_1, w_1)$ and $\mathbf{s}_2 = (u_2, v_2, w_2)$,

$$\int_\Omega \{(u_2 - u_1)_x + (v_2 - v_1)_y + (w_2 - w_1)_z\}^2\, d\Omega \geq 0 \tag{B.10}$$

Some manipulation of terms show that (B.10) is equivalent to the following.

$$\begin{aligned}
&\int_\Omega (u_{2x} + v_{2y} + w_{2z})^2 \, d\Omega \qquad \text{(B.11)} \\
&\geq \int_\Omega (u_{1x} + v_{1y} + w_{1z})^2 \, d\Omega \\
&\qquad + 2 \int_\Omega \{(u_{1x} + v_{1y} + w_{1z}) \\
&\qquad\qquad + [(u_{2x} - u_{1x}) + (v_{2y} - v_{1y}) + (w_{2z} - w_{1z})]\} \, d\Omega
\end{aligned}$$

The left hand side of (B.11) is $e_D(\mathbf{s}_2)$; and in view of (B.9), the right hand side works out to be $e_D(\mathbf{s}_1) + \delta e_(\mathbf{s}_1; \mathbf{s}_2 - \mathbf{s}_1)$. Therefore, by Lemma 3.5, e_D is convex on $\mathbf{C}^1(\Omega)$.

Appendix C

Partial List of Symbols

$\mathbf{r} = (x, y, z)$	:	spatial variables or coordinates
$\mathbf{R} = (X, Y, Z)$	:	material variables or initial position
t	:	time
$\mathbf{s}(\mathbf{r}, t) = (u, v, w)$	:	velocity in Eulerian Description
$\mathbf{S}(\mathbf{R}, t)$	:	velocity in Lagrangian Description
f	:	density image
$D/Dt\, g$	:	mobile derivative of g with respect to t
$g_x = \partial g/\partial x$	:	partial derivative of g with respect to x
$\nabla = (\partial/\partial x, \partial/\partial y, \partial/\partial z)$	:	gradient operator, spatial
$\nabla \cdot$	:	divergence operator
(T_x, T_y, T_z)	:	spatial extent of the imaging experiment
Ω	=	$[0, T_x] \times [0, T_y] \times [0, T_z]$, imaging volume
$\partial\Omega$	:	surface enclosing Ω

$d\Omega = dx\,dy\,dz$	:	differential volume element
e_S	:	cost pertaining to smoothness of $\mathbf{s} = (u, v, w)$
e_I	:	cost pertaining to incompressibility constraint
e_D	:	cost pertaining to divergence-free constraint
γ_1	:	regularization parameter pertaining to e_I
γ_2	:	regularization parameter pertaining to e_D
$\mathbf{f}$	:	discrete version of image f
$\mathbf{x}$	:	discrete version of the velocity field $\mathbf{s} = (u, v, w)$
Z	:	constraining space
Z^*	:	dual of the constraining space Z
η	:	usually an element of Z
η^*	:	an element of Z^*
ϕ	:	dual functional
δ_1	:	mean-squared error pertaining to e_I
δ_2	:	mean-squared error pertaining to e_D

References

[1] D. P. Boyd, *Comparative Cardiac Imaging: Function, Flow, Anatomy, and Quantitation,* B. Brundage, ed., ch. 3. Rockville, MD: Aspen Publishers, 1990.

[2] D. Marr, *Vision.* Boston, MA: Freeman, 1982.

[3] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems.* Washington, DC: Winston and Sons, 1977.

[4] J. Hadamard, *Lecture on the Cauchy Problem in Linear Partial Differential Equations.* New Haven, CT: Yale University Press, 1923.

[5] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[6] B. G. Schunck, "Image flow: fundamentals and future research," in *Proc. IEEE Conf. Comp. Vision and Patt. Rec.*, vol. CVPR-85, (San Francisco, CA), pp. 560–571, 1985.

[7] W. Enkelmann, "Investigations of multigrid algorithms for the estimation of optical flow fields in image sequences," *Comp. Vision, Graphics and Image Proc.*, vol. 43, pp. 150–177, 1988.

[8] H. Nagel, "On a constraint equation for the estimation of displacement rates in image sequences," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, no. 1, pp. 13–30, 1989.

[9] J. Aisbett, "Optical flow with an intensity-weighted smoothing," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, no. 5, pp. 512–522, 1989.

[10] N. Cornelius and T. Kanade, "Adapting optical flow to measure object motion in reflectance and x-ray image sequences," in *ACM Siggraph/Sigart Interdisplinary Workshop, Motion: Representation and Perception*, (Toronto, Canada), pp. 50–58, 1983.

[11] G. E. Mailloux, A. Bleau, M. Bertrand, and R. Petitclerc, "Computer analysis of heart motion from two-dimensional echocardiograms," *IEEE Trans. Biomed. Engr.*, vol. BME-34, no. 5, pp. 356–364, 1987.

[12] G. E. Mailloux, F. Langlois, P. Simard, and M. Bertrand, "Restoration of the velocity field of the heart from two-dimensional echocardiograms," *IEEE Trans. Med. Imaging*, vol. 8, no. 2, pp. 143–153, 1989.

[13] D. C. Youla and H. Webb, "Image restoration by the method of convex projections: Part 1: Theory," *IEEE Trans. Med. Imaging*, vol. 1, no. 2, pp. 81–94, 1982.

[14] A. Chorin and J. Marsden, *A Mathematical Introduction to Fluid Mechanics.* New York: Springer-Verlag, 1979.

[15] S. C. Hunter, *Mechanics of Continuous Media.* New York: John Wiley and Sons, 2nd ed., 1983.

[16] L. A. Segel, "An introduction to continuum theory," in *Modern Modeling of Continuum Phenomena* (R. C. DiPrima, ed.), pp. 1–60, American Mathematical Society, 1986.

[17] J. M. Fitzpatrick, "A method for calculating fluid flow in time dependent density images," in *Proc. IEEE Conf. Comp. Vision and Patt. Rec.*, vol. CVPR-85, (San Francisco, CA), pp. 78–81, 1985.

[18] J. M. Fitzpatrick, "The existence of geometrical density-image transformations corresponding to object motion," *Comp. Vision, Graphics and Image Proc.*, vol. 44, pp. 155–174, 1988.

[19] G. Meier, M. Ziskin, W. Santamore, and A. Bove, "Kinematics of the beating heart," *IEEE Trans. Biomed. Engr.*, vol. BME-27, no. 6, pp. 319–329, 1980.

[20] K. R. Walley, M. Grover, G. Raff, W. Benge, B. Hannaford, and S. Glantz, "Left ventricular dynamic geometry in the intact and open chest dog," *Circulation Research*, vol. 50, pp. 573–589, 1982.

[21] C. K. Kim, G. M. Min, M. M. Lee, J. D. Seo, Y. W. Lee, and M. C. Han, "Estimation of local cardiac wall deformation and regional wall stress from biplane coronary cineangiograms," *IEEE Trans. Biomed. Engr.*, vol. BME-32, no. 7, pp. 503–512, 1985.

[22] K. Subbaraj, D. N. Ghista, and E. L. Fallen, "Intrinsic indices of the left ventricle as a blood pump in normal and infarcted left ventricles," *J. Biomed. Eng.*, vol. 9, pp. 206–215, 1987.

[23] H. Azhari, S. Sideman, R. Beyar, E. Grenadier, and U. Dinnar, "An analytical descriptor of three-dimensional geometry: application to the analysis of the left ventricle shape and contraction," *IEEE Trans. Biomed. Engr.*, vol. BME-34, pp. 345–355, 1987.

[24] A. Amini, R. Owen, P. Anandan, and J. Duncan, "Non-rigid motion models for tracking the left-ventricular wall," in *Information Processing in Medical Imaging '91*, (Wye, UK), pp. 343–357, Springer-Verlag, 1991.

[25] E. A. Zerhouni, D. M. Parish, and W. J. Rogers, "Human heart: tagging with MR imaging—a method for non-invasive assessment of myocardial motion," *Radiology*, vol. 169, pp. 59–63, 1988.

[26] L. Axel and L. Dougherty, "MR imaging of motion with spatial modulation of magnetization," *Radiology*, vol. 171, pp. 841–845, 1989.

[27] L. Axel and L. Dougherty, "Heart wall motion: improved method of spatial modulation of magnetization for MR imaging," *Radiology*, vol. 172, p. 349, 1989.

[28] J. Prince and R. McVeigh, "Motion estimation from tagged MR image sequences," Tech. Rep. JHU/ECE 91-05, The Johns Hopkins University, 1991.

[29] N. J. Pelc, L. R. Pelc, R. J. Herfkens, and G. H. Glover, "Measurement of myocardial motion dynamics with phase-contrast cine MR imaging," *Radiology*, vol. 177(P), p. 171, 1990.

[30] N. J. Pelc and M. A. Bernstein, "Reduce scan time with improved signal-to-noise ratio in phase-contrast flow MR imaging," *Radiology*, vol. 177(P), p. 171, 1990.

[31] Y. C. Fung, *Biodynamics: Circulation.* New York: Springer-Verlag, 1984.

[32] P. Y. Simard and G. E. Mailloux, "A projection operator for the restoration of divergence-free vector fields," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 10, no. 2, pp. 248–256, 1988.

[33] S. M. Song and R. M. Leahy, "Determining 3-D velocity fields from 3-D cine CT images," in *Conference Record of the IEEE Nuclear Science Symposium*, vol. 2, (Crystal City, VA), pp. 1141–1146, 1990. Presented at the IEEE Med. Imaging Conference, 1990.

[34] S. M. Song and R. M. Leahy, "Computation of 3-D velocity fields from 3-D cine CT images of a human heart," *IEEE Trans. Med. Imaging*, vol. 10, no. 3, pp. 295–306, 1991.

[35] J. M. Fitzpatrick and C. A. Pedersen, "A method for calculating velocity in time dependent images based on the continuity equation," in *Proc. Electronic Imaging '88*, pp. 347–352, 1988.

[36] R. P. Feynman, R. B. Leighton, and M. Sands, *The Feynman Lectures on Physics*, vol. 2. Menlo Park, CA: Addison-Wesley, 1964.

[37] M. Sugawara *et al.*, *Blood Flow in the Heart and Large Vessels,* M. Sugawara *et al.*, eds., ch. 5, pp. 63–67. New York: Springer-Verlag, 1989.

[38] M. Bertero, T. A. Poggio, and V. Torre, "Ill-posed problems in early vision," *IEEE Proc.*, vol. 76, no. 8, pp. 869–889, 1988.

[39] A. Balakrishnan, *Applied Functional Analysis.* New York: Springer-Verlag, 2nd ed., 1981.

[40] E. Kreyszig, *Introductory Functional Analysis with Applications.* New York: John Wiley and Sons, 1978.

[41] I. Ekeland and R. Temam, *Convex Analysis and Variational Problems.* New York: American Elsevier Publishing Company, 1976.

[42] D. Lee, A. Papageorgiou, and G. W. Wasilkowski, "Computational aspects of determining optical flow," in *Proc. 2nd Int. Conf. on Comp. Vision*, (Tampa, FL), pp. 612–618, 1988.

[43] R. Courant and D. Hilbert, *Methods of Mathematical Physics*, vol. 1. New York: Interscience Publishers, 1953.

[44] G. D. Smith, *Numerical Solution of Partial Differential Equations: Finite Difference Methods.* Oxford: Oxford University Press, 1983.

[45] D. G. Luenberger, *Linear and Nonlinear Programming.* Menlo Park, CA: Addison-Wesley, 2nd ed., 1984.

[46] R. Fletcher, *Practical Methods of Optimization.* New York: John Wiley and Sons, 2nd ed., 1987.

[47] C.-C. J. Kuo and B. Levy, "Discretization and solution of elliptic PDEs—a digital signal processing approach," *IEEE Proc.*, vol. 78, no. 12, pp. 1808–1842, 1990.

[48] B. R. Hunt, "The application of constrained least squares estimation to image restoration by digital computer," *IEEE Trans. Computers*, vol. C-22, no. 9, pp. 805–812, 1973.

[49] A. K. Jain, "A sinusoidal family of unitary transforms," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 1, no. 4, pp. 356–365, 1979.

[50] A. R. Mitchell and D. F. Griffiths, *The Finite Difference Method in Partial Differential Equations.* New York: John Wiley and Sons, 1980.

[51] M. Hestenes, *Conjugate Direction Methods in Optimization.* New York: Springer-Verlag, 1953.

[52] L. Hageman and D. Young, *Applied Iterative Methods.* San Diego, CA: Academic Press, 1981.

[53] D. G. Luenberger, *Optimization by Vector Space Methods.* New York: John Wiley, 1969.

[54] D. P. Boyd and D. W. Farmer, *Cardiac Imaging and Image Processing,* S. M. Collins and D. J. Skorton, eds., ch. 4, pp. 57–87. New York: McGraw-Hill Book Company, 1986.

[55] A. Thompson, J. Brown, J. Kay, and Titterington, "A study of methods of choosing the smoothing parameter in image restoration by regularization," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, no. 4, pp. 326–339, 1991.

[56] C. C. Paige and M. A. Saunders, "Solution of sparse indefinite systems of linear equations," *SIAM J. Numerical Anal.*, vol. 12, no. 4, pp. 617–629, 1975.

[57] D. G. Luenberger, "Hyperbolic pairs in the method of conjugate gradients," *SIAM J. Applied Math.*, vol. 17, no. 6, pp. 1263–1267, 1969.

[58] B. K. P. Horn, *Robot Vision.* Cambridge, MA: The MIT Press, 1986.

[59] O. Axelsson and V. Barker, *Finite Element Solution of Boundary Value Problems: Theory and Computations.* New York: Academic Press, 1984.