Segmentation and Tracking of the Left Ventricle in Cardiac MRI

by

Leyla Imanirad

A thesis submitted in conformity with the requirements
for the degree of Master of Applied Science
Graduate Department of The Edward S. Rogers Sr. Department of
Electrical and Computer Engineering
University of Toronto

Abstract

Segmentation and Tracking of the Left Ventricle in Cardiac MRI

Leyla Imanirad

Master of Applied Science

Graduate Department of The Edward S. Rogers Sr. Department of Electrical and Computer Engineering

University of Toronto

2006

In this work, we have presented a novel approach to the problem of segmentation and tracking of the left ventricle (LV). A set of radial lines are defined which meet at roughly the centre of the LV. Intersection of each line with the myocardium, surrounding the LV, is estimated in a recursive Bayesian framework using particle filters. The probability distribution of boundary estimates on each line is calculated based on the directional derivative of the given line initially. This probability density is coupled with the data from adjacent lines to yield a better estimate of the boundary positions in an iterative process.

Acknowledgements

First and foremost I am indebted to my supervisor Dr. James MacLean for his incredible guidance, patience and understanding during my graduate life. I wouldn't have been standing here if it wasn't for his constant support. I am also very thankful to Dr. David Fleet for his gracious discussions and feedback regarding this project. My deepest gratitude to Dr. Perry Radau and Desmond Chung at Ontario Consortium for Cardiac Imaging (OCCI) for their participation in initial phases of this project and for providing all the cardiac data sets used in this project. I also like to acknowledge Ontario Graduate Scholarship (OGS) and U of T Fellowship for providing financial assistance through my master's studies.

My friends at "Vision & Image" lab have been the greatest source of encouragement and support both in academic and life matters. Thank you Joshua, Cristina, Divyang, Keir, and Siraj! I'd like to recognize Andre's assistance in analyzing some data for my thesis as well.

I have met some incredibly nice and intelligent people during my master's studies at U of T, naming them all would be a whole project by itself. Nonetheless, I'd like to name a few people whose company has always been an enjoyable experience for me. Daniel, Iqbal, Sam M, Bryan, Ahmad, Simon, Fernando, and Tom thank you all for sharing your ideas and all the great discussions.

I am deeply thankful to Matthijs, Sam H, Stephane, Maral and Hooman for all the food (hmmm chocolate), fun, rides and most important of all for being there when I needed them the most.

Coming to the end of my journey at University of Toronto, I am obliged to Professor Mike Carter, whose mentorship and support has had great influence on my life both inside and outside of the university.

At last but certainly not the least, I am very grateful to my family and friends outside of the university for their patience, understanding and compassion all the way through my student life.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

According to estimates from the American Heart Association [1], each year 17 million people die of cardiovascular disease (CVD) around the globe. More than 30% of all deaths in Canada are due to cardiovascular disease, an estimated 13% of which is caused by smoking. CVD is responsible for more than 75,000 deaths, claiming more lives than any other disease in Canada [2]. Based on Canadian Institutes of Health Research in 1998, with $18.5 billion in expenditures, CVD was the most costly disease, accounting for 11.6% of the total cost of illness in Canada. Therefore any effort to improve screening, diagnosis and treatment of CVD is highly beneficial to society.

Generally, cardiac examination involves assessing a combination of the four following physilogical measures: cardiac structure, function, perfusion and myocardial viability. Different imaging modalities, including Ultrasound (US), single-photon emission computer tomography (SPECT), computed tomography (CT), and magnetic resonance imaging (MRI) are used in performing cardiac examinations. Among existing methods, cardiac MRI (CMR) has attracted significant interest in the research community, as a single imaging technique capable of retrieving all cardiac measures [3, 4], appropriately given the name “one-stop shop”. Despite MRI’s effectiveness and wide acceptance in cardiovascular research, it is used infrequently in clinical applications due to some limitations. The patient’s ability to remain still plays a major role in quality of MR imaging, as patient movements and respiratory motion create artifacts in MR images, causing degradation of image quality. However, advancements in MRI hardware and acquisition systems have improved image quality by minimizing such effects. Additionally, each MRI acquisition consists of six to ten cross-sectional slices of the heart, all scanned in 20 phases over the cardiac cycle. Including some long-axis scans, there are about 200 images to be processed per cardiac acquisition session. Manual analysis of these images is an extremely time-consuming and laborious task at minimum, not to mention subjective and error-prone. Therefore, development of automatic systems to analyze these images and derive useful clinical information from them is highly desirable.

Analysis of the left ventricle (LV) in particular has attracted a lot of attention in the medical imaging community, as numerous CVD symptoms are manifested through variations in the left ventricle’s volume, mass or motion pattern. Ejection fraction of the left ventricle is another important clinical measure, which is derived based on the LV volume at two critical cardiac phases. While segmentation of the left ventricle in each short-axis slice at any time instance provides volumetric data of the given phase, tracking or detection of the ventricle boundaries through the cardiac cycle represent the ventricle motion.

Segmentation and/or tracking of the LV has been an active research topic for the past decade. Numerous vision-based algorithms have been utilized to address this problem, the most prevalent of which include thresholding, contour-based, and shape-based methods. Recently different formulation of probabilistic models have also been deployed to solve this problem. Most of these algorithms require either manual initialization of the contours in the first image (of a stacked data set), or they rely on a prior model obtained from training sets.

1.2 Thesis Objective

Our objective is to segment the LV in spatio-temporal sequences (2D+T) of short-axis cardiac MRI with minimal user interaction. The motion of left ventricle can be traced through time once segmentation results for each 2D scan are available. We do not require manual segmentation of the first frame or a training set for that matter[1]. The problem has been formulated in a probabilistic framework, and particle filters are used in estimating boundary positions. We make use of the fact that image data are causal, and combine image measurements of each frame with boundary estimation of the previous frame to perform online detection of the left ventricle boundary. The result is a fast and efficient algorithm that is comparable with other vision-based methods.

This thesis is organized as follows. Chapter 2 reviews relevant physiology, as well as a brief summary of MR imaging technology. The theory behind particle filters and Bayesian estimation frameworks is also discussed in Chapter 2. This chapter also entails a literature review of the methods that have been applied to the LV segmentation problem. Chapter 3 provides a detailed discussion of our design; representation of the problem in a probabilistic framework, estimation of boundaries at each frame, and propagation of boundaries from one frame to the next. Qualitative results of using this algorithm at different stages, and quantitative measures comparing our method with hand-segmented data are presented in Chapter 4. We conclude by providing a summary of this thesis, along with suggestions for future directions, in Chapter 5.

[1] Although such data could be used to improve the proposed method.

Chapter 2

Background

We introduced the LV segmentation problem as the subject of our work in the previous chapter. This chapter provides a basic overview of the background theory necessary for understanding the problem at hand, and the formulation of the proposed solution. Section 2.1 gives a summarized description of heart physiology and different physiological measures derived from CMR image analysis. The basis of the MR imaging modality, and its usage in diagnosis of cardiovascular disease, are discussed in Section 2.2. Section 2.3 contains a literature review of various methodologies that are proposed to address this problem. Finally, details of the Bayesian estimation frameworks and particle filters, which form the basis of our design, are discussed in Section 2.4.

2.1 Physiology Review

2.1.1 Heart Anatomy and Dynamics

The cardiovascular system consists of the heart and two vascular systems- the systemic and pulmonary circulations [5]. The human heart, shown in figure 2.1, is a muscular four-chambered organ, responsible for pumping blood throughout the body using rhythmic actions, or what is commonly known as the heart beat. The top two chambers, or atria, work as a reservoir to collect the blood that is coming into the heart. The bottom two chambers, or ventricles, have strong muscles, enabling them to provide the pumping action required to push the blood out of the heart and through the vascular systems. The right side of the heart is part of the pulmonary circulation, which receives de-oxygenated blood from body organs and delivers it to the lungs. The left side, on the other hand, is an essential part of the systemic circulation and collects the oxygenated blood in the left atrium and pumps it to various body organs using the left ventricle. Since a larger force is required from the left side to participate in systemic circulation, the left ventricle has a much thicker and more muscular wall. The *myocardium*, depicted in Figure 2.1, is the muscular wall of the heart. The smooth outer surface of the myocardium is called the epicardium, while the inner lining is known as the endocardium. The apex and base of the heart, as one might expect are the bottom and the top of the heart as displayed in Figure 2.1.

Each cardiac cycle is divided into two main stages [6]:

- Diastole: during the period of diastole (or relaxation), the atria contract in order to push the blood to ventricles, which are relaxed and filling with blood.

- Systole: during this period the ventricles contract to pump the blood out of the heart to lungs and other body tissues, while the atria are relaxed and filling.

Considering the fact that normal heart rate of a healthy adult is around 72 bpm (beats per minute), each cardiac cycle takes approximately 0.8 seconds to complete.

2.1.2 Physiological Measurements

General health and proper function of the heart depends on numerous factors. Evaluation of cardiac function is performed based on global and local physiological measurements. Some of these measurements are listed below:

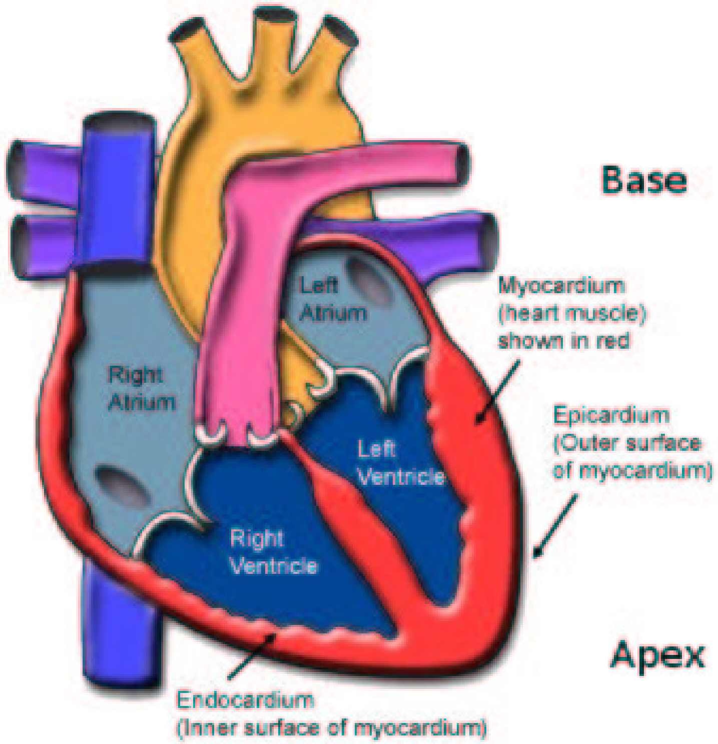

Base
Left Atrium
Right Atrium
Myocardium (heart muscle) shown in red
Epicardium (Outer surface of myocardium)
Left Ventricle
Right Ventricle
Apex
Endocardium (Inner surface of myocardium)

Figure 2.1: Heart Structure (image reproduced from [7])

- End-Diastolic Volume (EDV): the volume of blood in a ventricle at the end of diastole, or filling stage.

- End-Systolic Volume (ESV): the volume of blood in a ventricle right after systole, or ejection period, which is a measurement of adequacy of cardiac ejection.

- Stroke Volume (SV): the volume of blood ejected from a ventricle at each heart beat, which is calculated based on EDV and ESV as follows:

$$SV = EDV - ESV. \tag{2.1}$$

- Ejection Fraction (EF): defined as

$$E_f = \frac{SV}{EDV} = \frac{EDV - ESV}{EDV} \tag{2.2}$$

is the fraction of the end-diastolic volume (EDV) that is ejected with each heart beat.

Ejection fraction (EF) is normally computed for the left ventricle unless otherwise stated. In addition to the above-mentioned measures, myocardial wall motion and thickness are also derivable from cardiac imaging. These physiological values assist in diagnosis of various heart conditions such as ischemia, or acquired heart disease [3].

2.2 MR Imaging

Many imaging modalities are used in the screening and diagnosis of cardiovascular disease. Some of the more established techniques, reviewed in [4], include: Ultrasound (US), single-photon emission computer tomography (SPECT), computed tomography (CT), and magnetic resonance imaging (MRI). Among these methods MRI deserves special attention as the ideal "one-stop shop", which can potentially address most cardic examination needs [4, 8]. The basic theory behind MRI and its applications in cardiac imaging are presented in the following section.

2.2.1 MRI Techniques

Magnetic resonance imaging (MRI) works based on detection of protons inside hydrogen nuclei (H), which are abundant in water and fat tissues inside the body [8]. An important characteristics of these protons is the *spin* property, which can be thought of as a small magnetic field. Once placed in a strong magnetic field, the magnetic moments associated with nuclear spins tend to align themselves with the magnetic field of the MRI scanner. These protons are excited using an oscillating magnetic field, called radio frequency (RF) field. The excitation results in the magnetization being tipped into the plane perpendicular to the main magnetic field. The energy that is released upon relaxation of these protons is detected, and the resulting signals are used to construct clinically useful images [9, 10].

Advancements in the field of MR imaging have led to an increase in its usage in clinical applications, especially in CMR imaging. Cardiac examination normally involves assessing a combination of the following measures [3]:

- Cardiac structure: used in visualizing cardiac morphology in order to identify abnormalities caused by congenital heart defects, right ventricle, or pericardial problems.

- Cardiac function: used in objective quantification of ventricular volumes and ejection fraction (EF) as defined in Section 2.1.2.

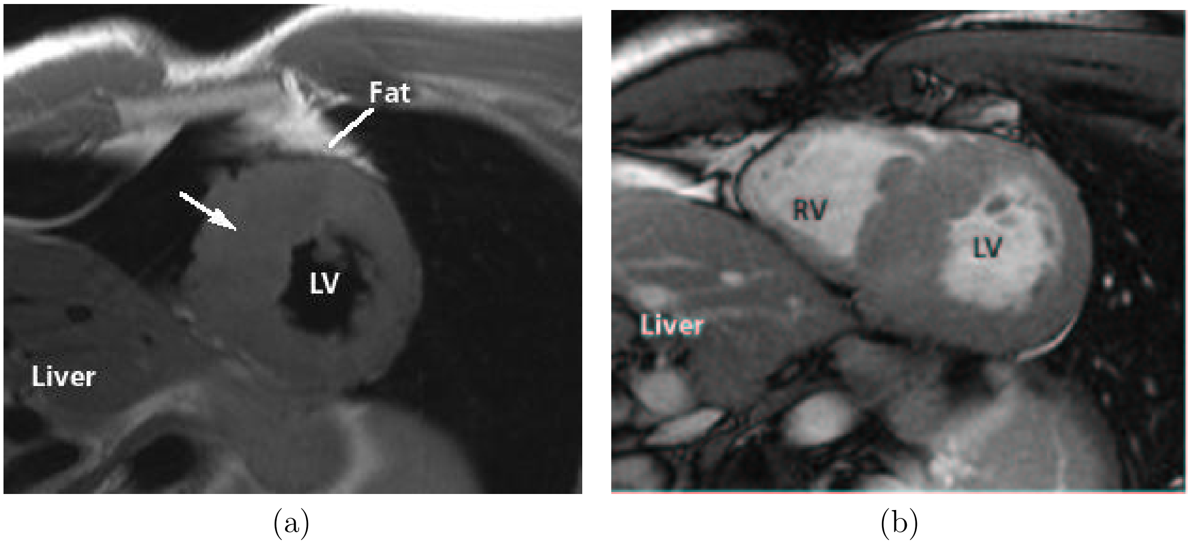
- Myocardial perfusion: is performed in the evaluation of the blood flow to the heart at rest and stress conditions. Radioactive contrast agents are used to trace distribution of blood to different areas of the heart.

- Myocardial viability: provides distinction between reversible and irreversible myocardial injuries. It plays an important role in prognosis of patients with coronary artery disease.

The focus of our work is on the first two measures, cardiac morphology and function. Depending on the specific application, different MRI acquisition methods are utilised.

Cardiac structure

The two basic techniques in CMR, which are used in accurate depiction of morphology, are "black-blood" and "bright-blood" methods. Spin-echo (SE) was the first method generating black-blood sequences which makes the blood appear darker than the myocardium and surrounding fatty tissues. This method has proven to be useful in visualizing morphology in congenital heart disease and pericardial abnormalities [8]. On the other hand, blood generates bright signal intensity, appearing lighter than myocardium, in bright-blood imaging [8]. The dominant sequences of this method include gradient-recalled echo (GRE) and fast GRE (fGRE). These methods yield both morphological and functional data as well as myocardial perfusion. A short-axis MRI scan using each of these methods is depicted in Figure 2.2.



(a)

(b)

Figure 2.2: (a) Short-axis spin-echo, (b) Short-axis gradient-recalled-echo (images reproduced from [10]).

Cardiac Function

Evaluation of cardiac function is performed in a global scale through analysis of EF, and in regional scale by measuring contractile disfunction. The latter is derived based on analysis of wall thickening in tagged MRI images, while the former is calculated based on cine MRI techniques. Cine MRI generates images of a single or multiple slices every 30–50 ms during cardiac cycle, providing temporal as well as spatial information. Since blood flow and respiratory motion create artifacts and noise in the resulting images, the imaging time of each scan should be minimized to avoid motion blurring. These artifacts pose some challenges in the analysis of cardiac images using traditional computer vision algorithms. Additionally, during each MRI acquisition, typically six to eight cross-sectional slices are acquired from the apex to the base, all with around 20 time phases over the entire cardiac cycle (one scan every 50ms) resulting in approximately 160 short-axis 2D scans of the heart [4]. Figure 2.3 depicts six slices from the apex to the base. As one can imagine, manual analysis of this 3D+T, or better said 4D dataset, is prohibitive and exhaustive, not to mention error-prone and subjective. This expensive and lengthy process certainly poses problems for widespread use of CMR, despite its great capabilities and potential. The need for semi- or fully-automatic methods to extract useful clinical measurements from this data has led to an extensive amount of research in the field of cardiac imaging. Segmentation of the left ventricle can be thought of as the first step in the derivation of ventricular volume or EF. The next section presents a summary of some

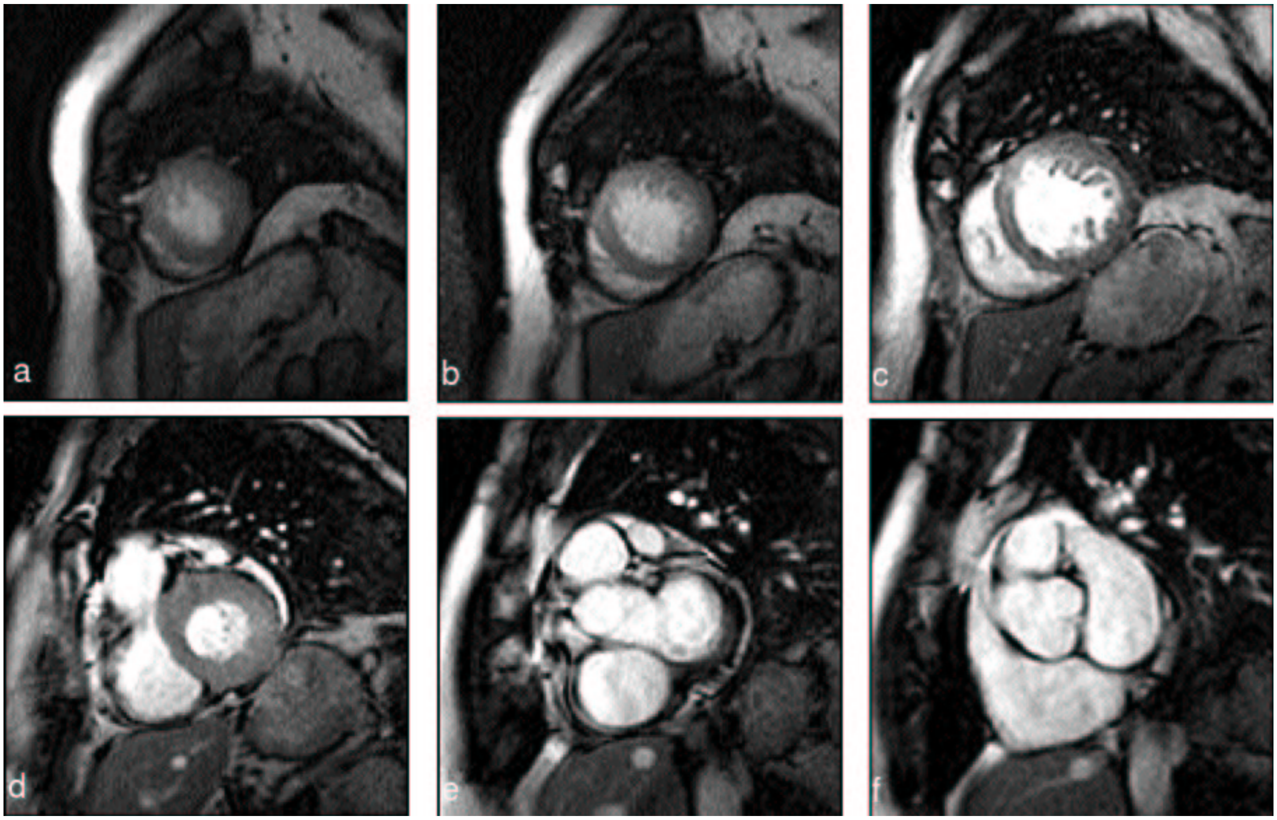


Figure 2.3: Six short-axis slices of heart from apex to the base using bright blood technique (image reproduced from [8])

computer vision methods and how they have been adapted to address this particular problem.

2.3 Literature Review

Numerous methods for segmentation and/or tracking of cardiac MRI in general, and left ventricle in particular, have been proposed in the past decade. A collection of these techniques are reviewed in [11]. In this section, we present a summary of these methods, categorized based on the fundamental computer vision algorithm applied.

2.3.1 Thresholding Methods

Higgins *et al.* [12] devised a semi-automatic method for segmentation of the left ventricle in 3D cardiac CT images. They utilise edge-preserving smoothing filters to reduce random noise while maintaining sharp edges and image features. Sanchez-Ortiz *et al.* [13] employed a thresholding technique to detect LV boundaries in 2D slices of 3D cardiac ultrasound (US). A fixed threshold value is set based on image intensities while a connected-component algorithm is executed on the 3D surface to identify the LV boundary. Lynch *et al.* [14] have also utilised intensities in a threshold clustering technique. These methods are efficient and easy to implement, however they are susceptible to intensity variations and low gradients associated with most medical images. At each image point, the gradient is defined as a 2D vector with components representing intensity derivatives in horizontal and vertical directions. Large intensity changes at a boundary result in large gradient magnitudes at the points on the boundary. Having considerable intensity variations at boundaries in an image is one of the main requirements of the edge-based methods, which poses a challenge for CMR, which does not offer great variations in intensity values between various tissue types.

2.3.2 Contour-based Methods

Originally formulated by [15], an active contour, also known as *snake*, is a specific type of deformable model. A snake is a deformable open or closed curve, often represented by a set of control points. A snake glides towards image features while maintaining continuity and smoothness of the curve. Using edges as image features to attract the active contour is a common approach, although the choice of features is application-specific [16]. The energy of the contour, $E(s)$, expressed as a function of control point indices s is defined as [17]:

$$E(s) \;=\; \sum_{s} \left(\underbrace{\alpha E_{continuity}(s) + \beta E_{curvature}(s)}_{\text{internal forces}} + \underbrace{\gamma E_{image}(s)}_{\text{image-based force}} \right) \tag{2.3}$$

The snake is deformed by minimizing a pre-defined energy function based on internal and external forces. While internal forces control continuity and smoothness of the contour, external forces drive the contour towards image features. Numerous segmentation methods for cardiac MRI have been developed based on active contours. Terzopolouos and McInerney [18] along with Davis [19] provide a detailed report on application of deformable models in medical imaging. In the case of 3D cardiac data, the snake is manually initialized by an expert, using a priori knowledge of the shape constraints at the segmentation problem. A user's interaction may be needed to fine-tune the curve in areas of high curvature. A good initial estimate for one slice is used as an initialization for the next slice in the sequence. Therefore, good initialization is required to obtain good segmentation performance. Spreeuwers & Breeuwer [20] propose to estimate position of epi- and endocardial contours jointly using active contours. A set of concentric radial lines are defined which meet approximately at the centre of the LV. The intensity profile along a radial line from the centre to the end of the line is derived. The active contour nodes are assigned to points that have significant intensity variation along this profile as shown in Figure 2.4. The transitional intensity model is initialized for all the radial lines of the first frame. This model is propagated through a time series, and the reference profile is translated and scaled to adjust to new contours at each time frame. Joint-estimation of contours makes this method more robust to low-contrast boundaries. We employ the idea of joint contour estimation, as well as using radial lines in our design.

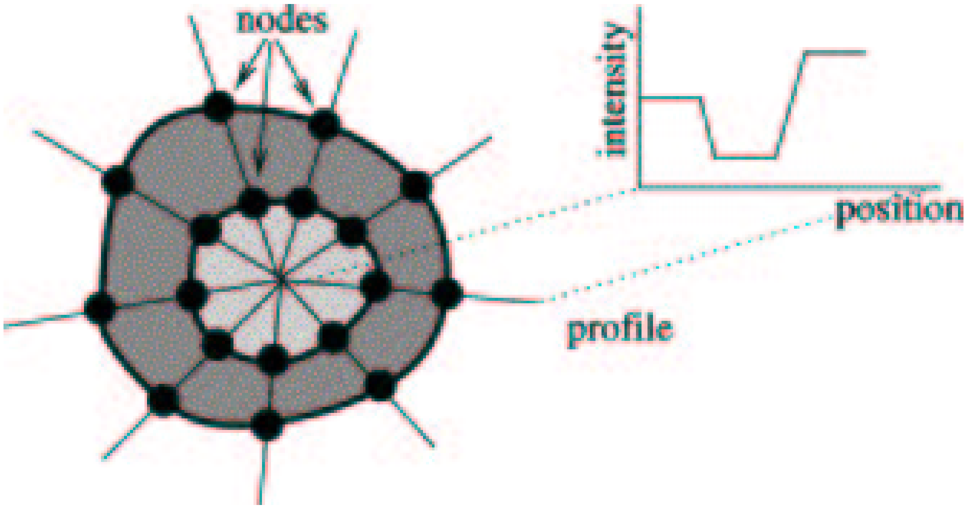


Figure 2.4: Intensity profiles used in joint boundary detection (image reproduced from [20])

Level-set methods, initially introduced by Osher and Sethian [21], have been utilised in numerous areas of computer vision including medical imaging. Level-sets are implicitly defined contours which evolve in time based on intrinsic geometric measures of the image and some speed function. The curve in 2D is defined as an isocontour of a 3D surface, i.e. a level-set. Due to their dynamic nature, these curves are capable of adapting to different topologies; so connectivity of contour regions is not a limiting factor in curve evolution [22]. Additionally, contour curvature at each point is utilized to impose smoothness constraints on the displacement of each point. These characteristics make level sets a promising approach in contour tracking. Caselles *et al.* [23] regard the boundary detection problem as finding a curve of minimal weighted length, or a *geodesic active contour*, in the level set framework. This is an iterative method which converges once the contour reaches a steady state. For high quality images with high-contrast boundaries, this technique yields good results for the LV segmentation problem. However this method is slow with high computation cost, and the convergence criteria may not be met if the boundary is low-contrast or non-existing.

2.3.3 Model-based Methods

One of the shortcomings of deformable models, discussed in Section 2.3.2 is that they are attracted towards high gradient edges, which causes problems in data frames with a lot of spurious edges. In order to make segmentation more robust to this noisy environment, prior knowledge is used to impose restrictions on the shape of the object. Active Shape Models (ASMs) [24], address the problem of locating objects with known shapes. Hand-annotated segmentation of an aligned set of cardiac data is used as the training set required in this method. Principle Component Analysis (PCA) is applied to represent the training set as a mean shape and common modes of shape variation. A global search is performed on a test image to match the mean shape to image features. This shape is then deformed based on the model's variation modes to better capture local characteristics of the object in the test image. Active Appearance Models (AAMs) [25] follow the same idea, with the exception that appearance characteristics such as intensity values or textures are used to construct the training set. Different research groups [26, 27, 28, 29] have applied either ASMs, AAMs, or a combination of both to the problem at hand. Considering the fact that the curve can only deform according to shape variations in the model, having a comprehensive model is key to the accuracy of these approaches. One of the disadvantages of such models is the manual segmentation required for the development of the training set which, in addition to being subjective, is a cumbersome and labour-intensive task. Additionally, the preprocessing step needed to perform PCA and align the test image to the training set increases the computational time of the algorithm.

2.3.4 Probabilistic Methods

In order to avoid the overhead introduced by the generation of a training set in model-based methods, probabilistic methods have been proposed to tackle the segmentation problem. Dias *et al.* [30] derive an estimation for position and thickness of the myocardium in sequences of echocardiographic images. A likelihood function is derived based on radial scan lines as previously seen in [20]. However in this case, the epi- and endocarial borders are derived disjointly. Additionally, the assumption is that three distinct regions along each scan line are identified: (a) centre point to endocarium, (b) myocardium, (c) outer myocardial boundary to the end of scan line. Maximum *a posteriori* (MAP) criteria is applied in finding the borders. Although the focus of this work is on echocardiographic images, the algorithm is also applicable to MRI, and we employ some of these ideas in our design. Weng [31] also uses intensity values in learning-based ventricle detection in cardiac MR images. Critical points along each scan line's intensity histogram are identified and used in maximum-likelihood (ML) estimation. In both of the above-mentioned techniques, the assumption is that intensity is roughly constant in each region, which does not necessarily holds true in cardiac data. Recent learning-based approaches like the ones proposed by Sun *et al.* [32], and Sénégas *et al.* [33] are more inclined towards using shape and motion models in a Bayesian framework. Both methods rely on estimations at each frame of a sequence to predict the boundaries in subsequent frames in a sample-based approach; however, they both require constructing an initial training set to be used for the shape model. Therefore, in order to achieve good segmentation results with these techniques, a large number of samples should be used to capture a wide variety of shape deformations, adding to the computational cost of the algorithm.

2.3.5 Thesis Contribution

In this thesis, we present a novel approach in segmentation and tracking of the left ventricle in a probabilistic framework. We have developed an algorithm to estimate inner and outer boundaries of the left venctricle on a set of radial lines, similar to the approach taken by Dias *et al.* [30]. The estimations for each line are propagated to the next line using particle filters. Motivated by the idea of joint boundary derivation using coupled active contours [20], the initial estimated boundaries from all the lines are refined in an iterative step. Finally, the final estimate of each frame is propagated to the subsequent frame, resulting in full segmentation of a 2D+T dataset.

2.4 Probabilistic Framework

In this section, we introduce the probabilistic framework which is applied in many computer vision problems. In the field of computer vision, where 2-D image data are the main source of information, inference of scene data from noisy images is a challenging task. Bayesian approaches combine the image data and prior knowledge about scene characteristics to infer the scene data, often referred to as state information.

2.4.1 Bayesian Formulation

The idea behind Bayesian approaches is to deploy prior knowledge about the scene to infer state information from observation (image) data. For the sake of formulation, observation data is defined as Z whereas the state is represented as X. The objective of using a Bayesian framework is to compute the conditionl probability $p(X|Z)$, frequently referred to as the *posterior probability distribution* in the literature. The *prior*, $p(X)$, embeds prior knowledge about the scene or state vector. Based on Bayes theorem:

$$p(X|Z) = \frac{p(Z|X)p(X)}{p(Z)}. \tag{2.4}$$

Since $p(Z)$ is derived based on image data and is independent of state information, it is treated as a normalizing factor resulting in a posterior:

$$p(X|Z) \propto p(Z|X)p(X). \tag{2.5}$$

Computation of $p(X|Z)$ given $p(X)$ and $p(Z|X)$ is called Bayesian inference, where the distribution $p(Z|X)$ is appropriately called the *likelihood function*, representing the likelihood that the state X generated the observed data Z.

2.4.2 Recursive Bayesian Framework

As described in the previous section, the Bayesian framework can be established to provide information regarding scene data. The same process is applicable to sequences of images recursively. Using this approach, the posterior probability distribution from the previous iteration serves as the prior for the current iteration.

A tracking problem in computer vision can be thought of as estimating state values from noisy observations or measurements. The state is normally a vector of multiple variables and it is defined based on the application. For instance, in our design we define the state as a two-dimensional vector, containing the parameters of boundary positions to be determined. In tracking applications the state to be determined is usually the trajectory, velocity, or position of moving objects. Observations, on the other hand, depend on the image data.

Let us first introduce the unknown state vector $\mathbf{x}_t$ as a n-dimensional vector containing variables to be estimated. Additionally, state history is defined as $\mathcal{X}_t = \{\mathbf{x}_1, \cdots, \mathbf{x}_t\}$ which represents the sequence of target states up to time t. Similarly the set of image features or observations from which we estimate the state is defined as $\mathbf{Z}_t$, with observation history represented by $\mathcal{Z}_t = \{\mathbf{Z}_1, \cdots, \mathbf{Z}_t\}$.

The goal of probabilistic formulation is to estimate the posterior distribution given observations, $p(\mathcal{X}_t|\mathcal{Z}_t)$. As defined by [34], the *marginal posterior*, or *filtering distribution*, at the current time is given by:

$$p(\mathbf{x}_t|\mathcal{Z}_t) = \int_{\mathbf{x}_1} \cdots \int_{\mathbf{x}_{t-1}} p(\mathcal{X}_t|\mathcal{Z}_t). \tag{2.6}$$

Using Bayes' rule, the posterior distribution can be written as

$$p(\mathcal{X}_t|\mathcal{Z}_t) = \frac{p(\mathcal{Z}_t|\mathcal{X}_t)p(\mathcal{X}_t)}{p(\mathcal{Z}_t)}. \tag{2.7}$$

The posterior Equation 2.7 is reformulated as

$$p(\mathcal{X}_t|\mathcal{Z}_t) \propto p(\mathcal{Z}_t|\mathcal{X}_t)p(\mathcal{X}_t) \tag{2.8}$$

assuming that the observation history is independent of states. Assuming a first order Markov model for dynamic states, which only considers the immediate preceding state, results in:

$$p(\mathbf{x}_t|\mathcal{X}_{t-1}) = p(\mathbf{x}_t|\mathbf{x}_{t-1}) \tag{2.9}$$

which results in defining the sequence prior $p(\mathcal{X}_t)$ as

$$p(\mathcal{X}_t) = \Big(\prod_{j=2}^{t} p(\mathbf{x}_j|\mathbf{x}_{j-1})\Big)p(\mathbf{x}_1). \tag{2.10}$$

Additionally, if we assume conditional independence of observation data, the sequential likelihood can be written as

$$\begin{aligned} p(\mathcal{Z}_t|\mathcal{X}_t) &= p(\mathbf{z}_t|\mathbf{x}_t)p(\mathcal{Z}_{t-1}|\mathcal{X}_{t-1}) \\ &= \prod_{k=1}^{t} p(\mathbf{z}_k|\mathbf{x}_k). \end{aligned} \tag{2.11}$$

Substituting for prior and likelihood terms from Equations 2.10 and 2.11 in the posterior distribution relation defined in Equation 2.8 results in

$$
\begin{aligned}
p(\mathcal{X}_t|\mathcal{Z}_t) &\propto p(\mathcal{Z}_t|\mathcal{X}_t)p(\mathcal{X}_t) \\
&= \prod_{k=1}^{t} p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_1)\prod_{j=2}^{t} p(\mathbf{x}_j|\mathbf{x}_{j-1}) \\
&\propto p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathcal{X}_{t-1}|\mathcal{Z}_{t-1})
\end{aligned}
\tag{2.12}
$$

which is a recursive expression for the posterior distribution. The filtering distribution can also be represented recursively as

$$
\begin{aligned}
p(\mathbf{x}_t|\mathcal{Z}_t) &= \int_{\mathbf{x}_1} \cdots \int_{\mathbf{x}_{t-1}} p(\mathcal{X}_t|\mathcal{Z}_t) \\
&= cp(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathcal{Z}_{t-1}),
\end{aligned}
\tag{2.13}
$$

where c is a normalizing constant, and the prediction distribution is

$$
p(\mathbf{x}_t|\mathcal{Z}_{t-1}) = \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathcal{Z}_{t-1}). \tag{2.14}
$$

The recursive nature of Bayesian inference is advantageous in the sense that there is no need to store all previous data; the past posterior, in addition to current measurements, is sufficient to infer the current probability distribution. Considering the Bayesian formulation discussed above, the recursion occurs in two stages: prediction (propagation) and observation (measurement). As depicted in Figure 2.5, the prediction stage involves dynamic motion of the state, whereas observation stage computes the likelihood of observation data given a certain state.

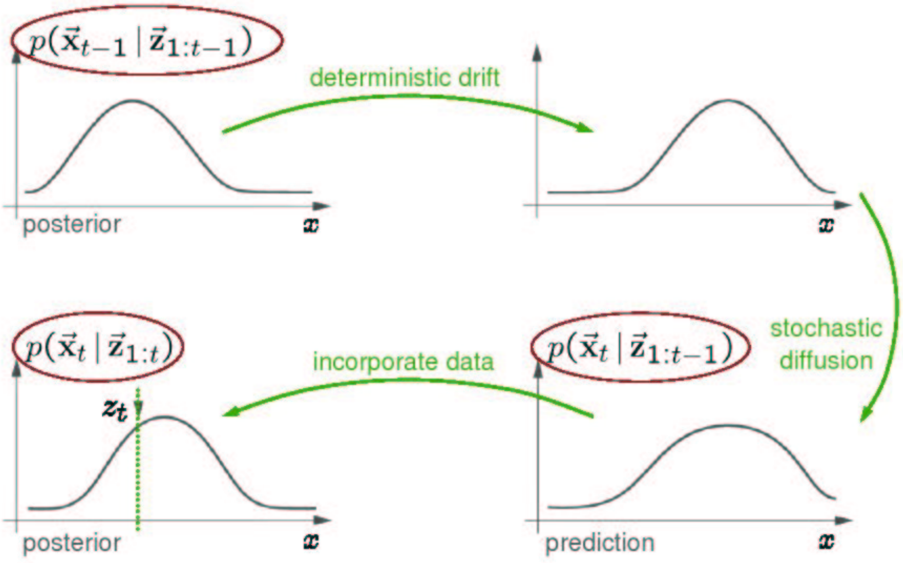

$p(\vec{x}_{t-1} \mid \vec{z}_{1:t-1})$
deterministic drift
posterior
x
x
stochastic diffusion
$p(\vec{x}_t \mid \vec{z}_{1:t})$
incorporate data
$p(\vec{x}_t \mid \vec{z}_{1:t-1})$
z_t
posterior
x
prediction
x

Figure 2.5: Recursive Bayesian Inference (image reproduced from [34]).

2.4.3 Particle Filter

In the previous section we established a probabilistic framework used in many computer vision applications. In this section we discuss the *particle filter*, a heuristic algorithm based on the Bayesian framework.

If the process is noise free then the states can be exactly determined based on the observational data. In reality, the observational data are almost never noise-free, and the posterior is estimated heuristically. If the posterior density at every time step is Gaussian, hence representable by its mean and variance, the recursive posterior density and filtering distribution in Equations 2.12 and 2.13 becomes a Kalman filter. In addition to the Gaussian density assumption, the Kalman filter also requires an assumption of linear dynamics to hold. Due to such restrictions practical applications of Kalman filters in non-Gaussian process estimations are limited. The Extended Kalman Filter (EKF) was developed to handle some of the short-comings of the Kalman filter [35]. Motion dynamics can be linear or non-linear in the EKF. It effectively approximates the non-Gaussian density as a Gaussian. If the true density is non-Gaussian, the approximation will be a poor representation of the underlying density function. In such cases, Sequential Monte Carlo (SMC) methods will yield better performance [36].

As defined by Ristic *et al.* [37], *Sequential Importance Sampling (SIS)* is a general Monte Carlo integration method that performs estimation based on a point mass or "particle" representation of probability densities. The key idea of a particle filter is to approximately represent the posterior distribution by a set of random samples and their associated weights. Better approximation to the density function can be achieved by increasing the number of samples or particles.

CONDENSATION Algorithm

Isard and Blake [38] first introduced the "CONDENSATION" algorithm in 1998. CONditional DENSity propagATION (CONDENSATION) algorithm aims to estimate state variables in sequences of images taken at successive times. The state of the object at time t is denoted as $\mathbf{x}_t$ with its history defined as $\mathcal{X}_t = \{\mathbf{x}_1, \cdots, \mathbf{x}_t\}$. Similarly, the image observations or measurements are denoted $\mathbf{Z}_t$ with history $\mathcal{Z}_t = \{\mathbf{Z}_1, \cdots, \mathbf{Z}_t\}$. The posterior distribution at iteration $t-1$ is represented by the weighted sample set $\mathbf{SS}_{t-1} = \left\{(\mathbf{s}_{t-1}^{(n)}, w_{t-1}^{(n)}), n = 1, \cdots, N\right\}$. The $\mathbf{s}_{t-1}^{(n)}$ value represents the state (position) of the n^{th} particle on the distribution at time $t-1$, whereas the weight, $w_{t-1}^{(n)}$, defines the likelihood of each state.

The prediction, following by the observation stage, discussed in Section 2.4.2 are applied to this sample set, resulting in posterior density estimation for the current iteration as shown in Figure 2.6. During the prediction stage, the particles are subjected to the motion model, which normally consists of a drift and a diffusion component as displayed in Figure 2.6. The likelihood function is applied to update the weights of the new particles based on $p(\mathbf{Z}_t|\mathbf{x}_t)$. In addition to the two steps mentioned above, resampling is performed to achieve more efficient sampling. The resampling stage involves selecting particles based on their weights; particles with higher weights may be selected multiple

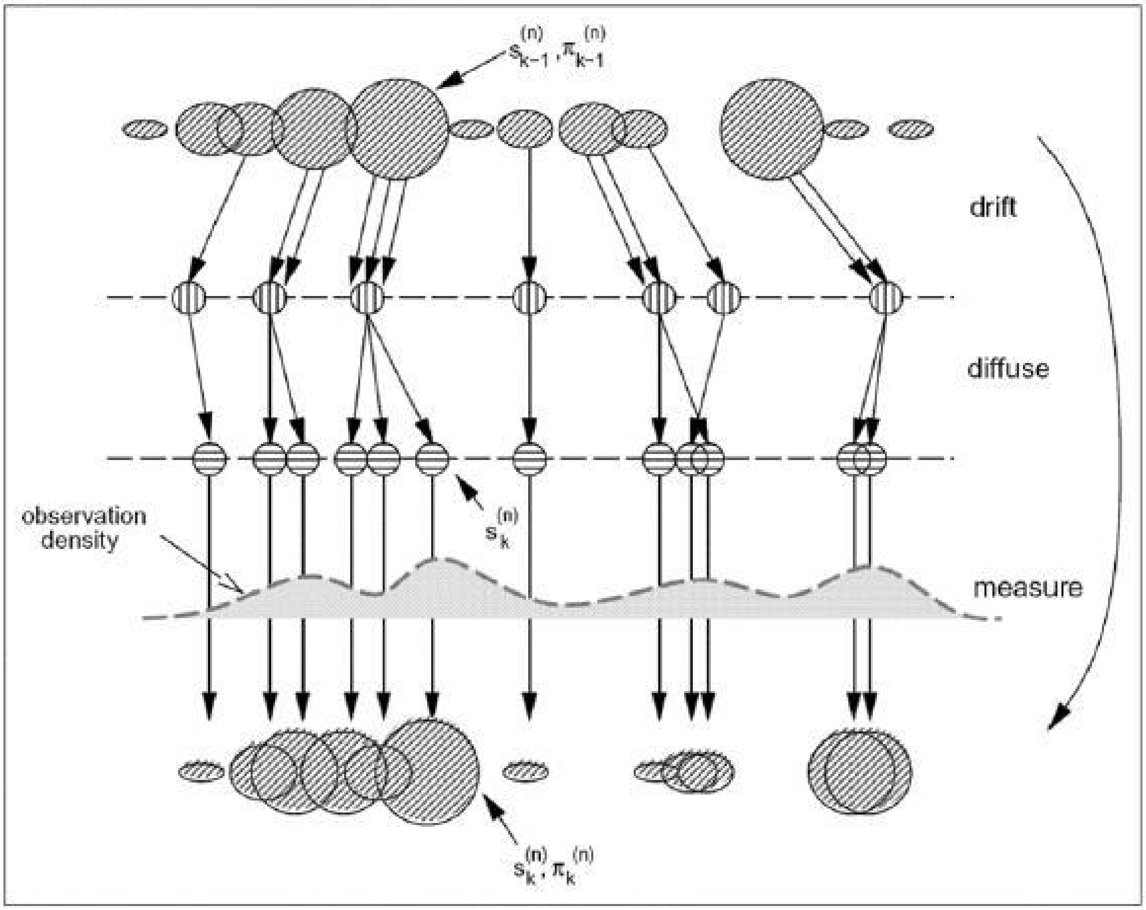


Figure 2.6: Particle Filter Progression (image reproduced from [38])

times, while particles with smaller weights may not be selected at all. The purpose of resampling is to ensure that there are enough particles to represent the distribution in the regions of high probability. In the next chapter we apply this framework in the segmentation of the LV in CMR images.

Chapter 3

Methodology

In this chapter we provide a detailed description of our algorithm used in segmentation and tracking of the LV. In order to do so, both endocardial and epicardial borders, also known as inner and outer borders, should be detected. Each sequence contains one slice of short-axis cardiac MRI in 20 phases. Each instance of a 2D scan is referred to as a frame, and these frames are captured at 50ms per frame. Strong temporal correlation between frames enables us to propagate the boundary estimation results from one frame to the next in a recursive Bayesian framework. In traditional approaches, the boundaries on the first frame are manually initialized. However, in our design we use particle filters in automatic estimation of the contours in the first frame with high efficiency and minimal user interaction. The design is decomposed into two main stages: (a) primary contour derivation (PCD), (b) iterative contour refinement (ICR). Figure 3.1 depicts a diagram of the overall system design. A set of radial lines are defined which meet roughly at the centre of the LV. The initial prior density is generated based on the user's selection of boundary points on the first line. The algorithm continues by propagation of sample sets from one line to the next in the same frame (spatially), or from a line to the same line in the next frame (temporally). Both the PCD, and ICR modules are implemented based on the particle filter algorithm, discussed in Section 2.4.3. Initially, particle filters are used in estimating boundary points on each line using image gradients as observation data. All lines are visited once and the intersection of each line with endocardial and epicardial boundaries are estimated. Once we have an initial estimate for every line, boundary positions are adjusted based on continuity and smoothness constraints imposed by neighbouring lines. The refinement step is iterative and probabilistic as well. In order to avoid systematic errors, the starting line and traversal direction– clockwise (CW) or counter clockwise (CCW)– for each iteration is chosen randomly.

3.1 System Initialization

The contour estimation problem is reformulated as estimating a set of points on each of the inner/outer boundaries and using spline fitting to derive the entire contours. A formal definition of radial lines and sample sets are discussed in this section.

3.1.1 Definition of Radial Lines

As part of the initialization step, the user is prompted to specify two points; the first point $O = [x_c, y_c]^T$ is selected roughly at the centre of LV while the second point, R, is chosen randomly outside the epicardium. Distance between the centre and the external point determines a radius ρ. We also define an angular sampling parameter $\Delta\omega$ as the angular separation between each two adjacent lines. Assuming polar coordinates centred at the origin O, each radial line in this system is represented by $l_i = [\rho, \theta_i]$ where $\theta_i = i * \Delta\omega$ for $i = 0, 1, \cdots, L-1$, where $L = \left\lfloor \frac{360}{\Delta\omega} \right\rfloor$ is the total number of radial lines in the image. This assumes that the initial line defines $\theta = 0$ in the defined coordinate system. For the images shown in this chapter $\Delta\omega = 15°$, resulting in $L = 24$.

In addition to line specification, intersection of the first line with the inner and outer wall, $\mathbf{x_p} = [r_p, d_p]$, are selected by the user. An example of these lines and boundary initializations on the first line are shown in Figure 3.2.

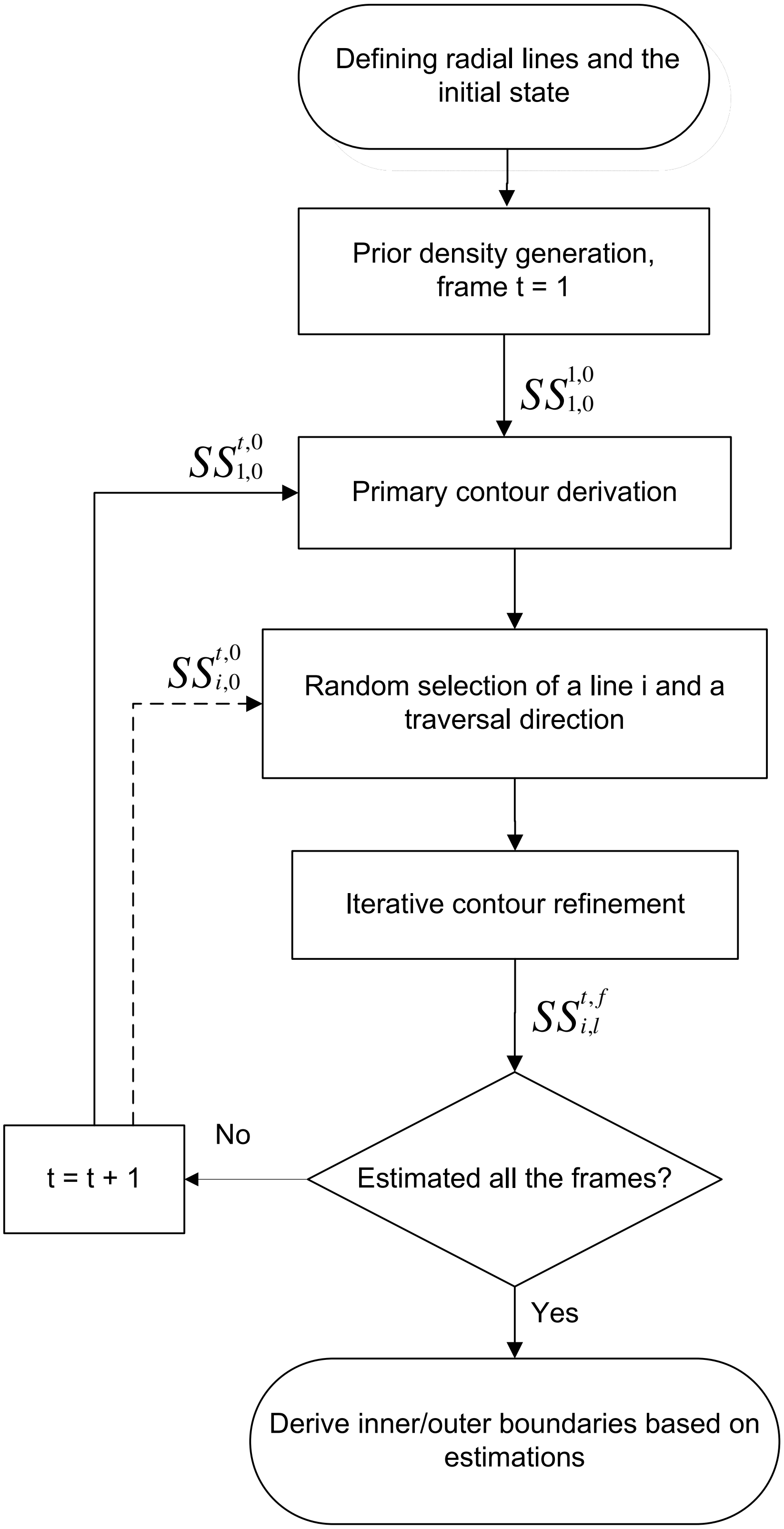

Defining radial lines and the initial state
Prior density generation, frame t = 1
$SS_{1,0}^{1,0}$
$SS_{1,0}^{t,0}$
Primary contour derivation
$SS_{i,0}^{t,0}$
Random selection of a line i and a traversal direction
Iterative contour refinement
$SS_{i,l}^{t,f}$
No
t = t + 1
Estimated all the frames?
Yes
Derive inner/outer boundaries based on estimations

Figure 3.1: High-level Design Flowchart

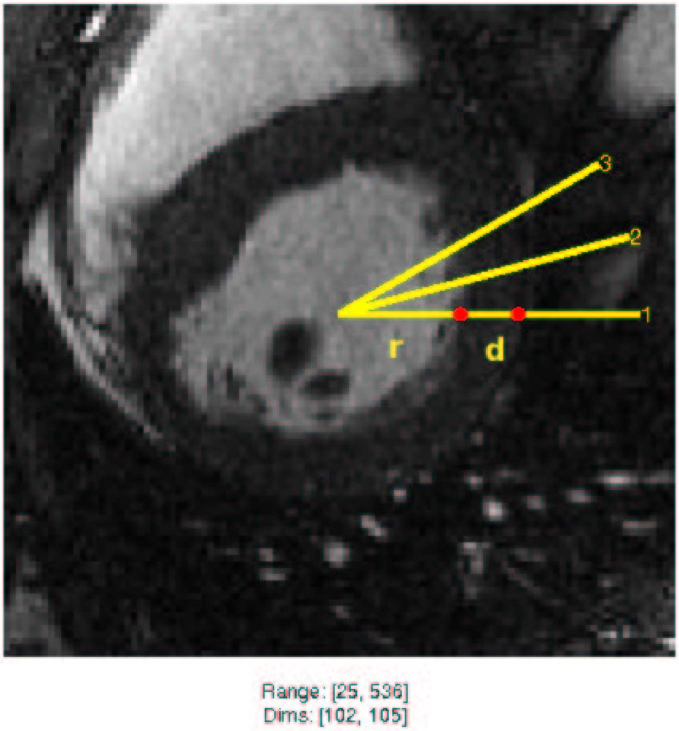


Figure 3.2: State Initialization on the First Line

3.1.2 Prior Density Generation

We covered the basis of the recursive Bayesian framework and particle filters in Section 2.4. The variable to be estimated is referred to as the state. For our design, we define the state to be $\mathbf{x} = [r, d]^T$ where r is the distance from centre to endocardial (inner) ventricle boundary, and d is the myocardium thickness (distance from the inner to the outer boundary). Equivalently, the state can be defined as $\mathbf{x} = [r, r']^T$ where $r' = r + d$ represents the distance from the centre to the outer boundary. Based on the specific application and ease of use, these two state definitions are used interchangeably throughout this document. We start with formal introduction of some of the notation used in our algorithm description. By definition, subscripts of sample set $\mathbf{SS}_{i,l}^{t,f}$, represent l^{th} iteration of line i, while superscripts denote f^{th} iteration of time frame t. A compact representation of sample set is denoted $\mathbf{SS_k}$ where $\mathbf{k} = (i, l, t, f)$. Additionally, $\mathbf{SS_k} = \left\{\mathbf{s_k}^{(j)}, w_{\mathbf{k}}^{(j)}\right\}$ with $\mathbf{s_k}^{(j)}$ denoting particle j^{th} and $w_{\mathbf{k}}^{(j)}$ representing weight of j^{th} particle at iteration $\mathbf{k}$.

The user-selected initial state $\mathbf{x_p} = [r_p, d_p]$, containing the initial state information of the first line, is used to construct the initial sample set, or prior density, of line one, $\mathbf{SS_{k0}}$, with $\mathbf{k}_0 = (1, 0, 1, 1)$. For the sake of simplification, $\mathbf{SS_{k0}}$ is represented as $\mathbf{SS_0} = \left\{\mathbf{s}_0^{(j)}, w_0^{(j)}\right\}$ for $j = 1, 2, \cdots, N$, where each particle $\mathbf{s}_0^{(j)} = \left[r_0^{(j)}, d_0^{(j)}\right]^T$ is constructed in two steps:

- the $r_0^{(j)}$ value is drawn randomly from a Normal distribution $\mathcal{N}(\mu_r; \sigma_r)$ with $\mu_r = r_p$ and parameter $\sigma_r = 5$ pixels, and

- the $d_0^{(j)}$ value is drawn randomly from a Rayleigh distribution $R(\alpha)$ with parameter α defined as $\alpha = d_p\sqrt{\frac{2}{\pi}}$.

Smaller values of σ_r create more localized distribution, whereas larger values of it will cause more divergence in the particle states. In order to ensure that values of d are always positive, we are randomly selecting them from a Raleigh distribution. The same Normal distribution used for r can be applied in deriving d as well, however that requires adding a verification step to make sure values of d never become negative. In addition to position states, we need to initialize weights for each particle. A uniform prior is assumed for initial weights resulting in $w_0^{(j)} = \frac{1}{N}$ for each sample $j = 1, 2, \cdots, N$. The total number of particles, N, is a user-specified parameter. Theoretically, increasing the number of particles will yield a better approximation to the true underlying density function. However, since computation time is directly influenced by the number of particles used, we limit ourselves to $N = 50$ in this specific application. More details on selection of N and how it effects the overall performance are discussed in the Chapter 4.

3.2 Primary Contour Derivation (PCD)

Starting from line $i = 1$, the posterior distribution is calculated for the line, using the three stages of the Condensation algorithm, as described in Section 2.4.3. Through an iterative procedure, diffusion of sample set $\mathbf{SS}_{k-1}$ results in new state positions. The posterior density, or sample set, $\mathbf{SS}_k$ is formed by calculating weights of the new particles based on the likelihood function. This new set is resampled, and used in the next iteration of the algorithm. The iteration for each line continues until the convergence criteria are met. The posterior distribution of line i, once converged, is used as the prior for line $i+1$, traversing in CCW direction. This process continues until all the lines are visited once. A flowchart, demonstrating sequences of events in this module, is depicted in Figure 3.3. A more detailed description of each stage is reviewed in this section.

3.2.1 Prediction

As depicted in Figure 2.5, the prediction stage involves two steps: a deterministic drift, and stochastic diffusion. Prior knowledge of some motion model determines the amount of drift from one iteration to the next, whereas stochastic diffusion includes inherent noise and uncertainties of the motion model. Whether prediction is performed on a single line or from one line to the next, there is no appropriate deterministic motion model known for our case. Therefore, our prediction stage only consists of stochastic diffusion, modeled using Gaussian noise. This approach is sometimes called a "random walk" motion model. Each particle $\mathbf{s}_{k-1}^{(j)} = \left[r_{k-1}^{(j)}, d_{k-1}^{(j)}\right]^T$, with weight $w_{k-1}^{(j)}= \frac{1}{N}$, in sample set $\mathbf{SS}_{k-1}$ is subjected to a prediction stage to give way to $\mathbf{s}_k^{(j)} = \left[r_k^{(j)}, d_k^{(j)}\right]^T$ as

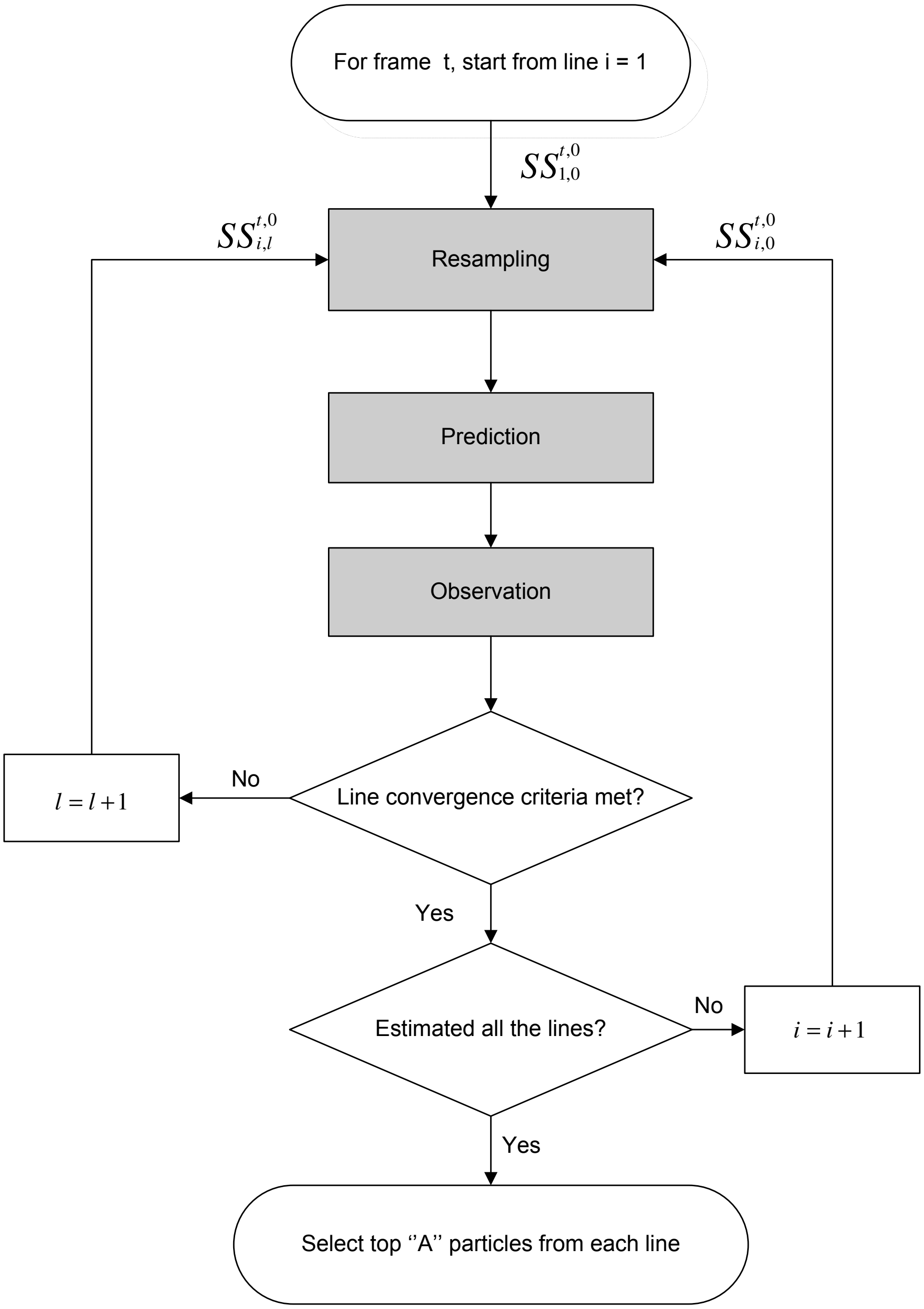


Figure 3.3: Primary Contour Derivation Design Flowchart

follows

$$
\begin{aligned}
\mathbf{s}_k^{(j)} &= \mathbf{s}_{k-1}^{(j)} + \mathbf{n}_{k-1} & \mathbf{n}_{k-1} = [\eta_r, \eta_d]^T \\
r_k^{(j)} &= r_{k-1}^{(j)} + \eta_r & \eta_r \sim \mathcal{N}(0; \sigma_r) \\
d_k^{(j)} &= d_{k-1}^{(j)} + \eta_d & \eta_d \sim \mathcal{N}(0; \sigma_d)
\end{aligned} \tag{3.1}
$$

where σ_r, and σ_d are user-defined parameters and are set to 5 and 3 pixels respectively. Selecting larger σ values creates more diffusion in the sample set.

3.2.2 Observation

The likelihood function $p(\mathbf{Z}|\mathbf{x})$ assigns weights to particles based on observation data $\mathbf{Z}$. The directional derivative D along each radial line serves as the underlying measurement used in this application. Intensity changes along a line result in peaks and troughs on the derivative D along that line. In order to calculate D_i along line $l_i = [\rho, \theta_i]$, we compute the directional derivative at each pixel (x, y) in the direction of θ_i as

$$D_i = I_x \cos(\theta_i) - I_y \sin(\theta_i), \tag{3.2}$$

where I_x and I_y are estimates of image gradients in the x and y directions respectively. These gradients are calculated by convolving the image with a-first-derivatives-of-Gaussian kernel in the x, and y directions. In order to preserve low-contrast edges, a small standard deviation of $\sigma_g = 2$ is selected for the Gaussian smoothing function. Recalling from Section 3.1.1, each line is identified by a θ and a ρ. In order to find the points on the line l_i, we transform the polar coordinates to the cartesian coordinates at evenly spaced values of ρ as

$$
\begin{aligned}
\bar{x} &= \rho \cos(\theta_i) \\
\bar{y} &= \rho \sin(\theta_i)
\end{aligned} \tag{3.3}
$$

where $x = \lfloor \bar{x} \rfloor$, and $y = \lfloor \bar{y} \rfloor$. To find the value of directional derivative at a specific radius corresponding to $(\bar{x}, \bar{y})$, we perform bilinear interpolation based on D_i as

$$\begin{aligned} D_{int,i} = D_i(x,y)(x+1-\bar{x})(y+1-\bar{y}) + D_i(x+1,y)(\bar{x}-x)(y+1-\bar{y}) + \\ D_i(x,y+1)(x+1-\bar{x})(\bar{y}-y) + D_i(x+1,y+1)(\bar{x}-x)(\bar{y}-y) \end{aligned} \quad (3.4)$$

For the sake of simplification, the interpolated derivative for the line i is denoted D_i. An example of a directional derivative, with $\Delta\rho = 1$ is depicted in Figure 3.4.

Derivative values with magnitude below a certain threshold, displayed as dashed lines in Figure 3.4, are surpressed. This threshold is initially set to 15% of $|D_i|_{max}$, although it is dynamically reduced to allow for less significant peaks, if deemed necessary. Since the transition from the ventricular cavity to myocardium is a bright to dark transition, the endocardial (inner) boundary intersection is assumed to be on a negative peak (trough). The outer boundary, on the other hand, could be on a negative or positive peak depending on whether it is adjacent to the right ventricle or to other organs. Observation data $\mathbf{Z}_i$ pertaining to line i is derived from derivative measurements of that line as the set

$$\mathbf{Z}_i = \left\{ [z, z']^T \quad s.t. \;\; z \in \mathbf{Z}^n, \quad z' \in \mathbf{Z}^a, \text{ and } z' \geq z + T_{min} \right\} \quad (3.5)$$

where $\mathbf{Z}^n$ is the set of negative peaks (troughs) in the derivative profile, and $\mathbf{Z}^a$ is the set of all peaks and troughs. The minimum allowed myocardium thickness, T_{min}, can be approximated from an end-diastole scan of a dataset, and is defined as 7 pixels in our application.

Once the observation set is constructed, particle weights are updated based on the likelihood function $w_i^{(j)} = p(\mathbf{Z}_i | \mathbf{x} = \mathbf{s}_i^{(j)})$ as derived in Section 2.4.3. This likelihood function is derived based on probabilities of all $M = |\{\mathbf{Z}_i\}|$ observation data given

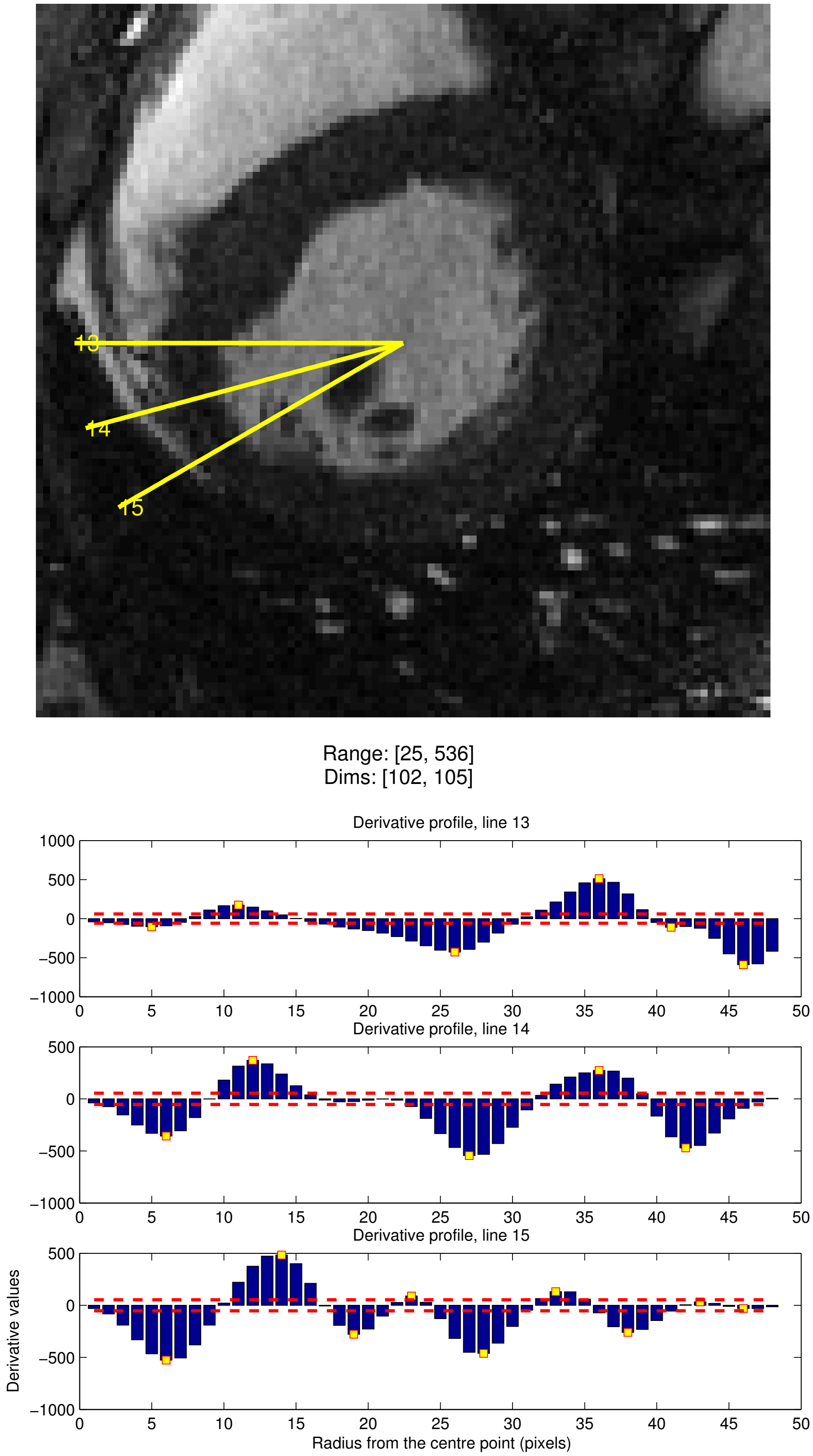


Figure 3.4: Directional Derivative of a Few Sample Lines

particle $\mathbf{x}$, and is calculated as

$$p(\mathbf{Z}_i|\mathbf{x}) = \gamma + \frac{1-\gamma}{M} \sum_{m=1:M} p(\mathbf{z}_m|\mathbf{x}) \tag{3.6}$$

where γ is included to account for outlier probabilities and dominates when observation data is weak or non-existing. Choosing an appropriate value for γ is a challenging problem, as the observation data vary from one line to the next. Therefore, we are setting $\gamma = 0$ in our experiments. Instead, in the case of weak observation data, we assign uniform weights to the particles.

Observation density for each measurement $\mathbf{z}_{m,i} = [z_m, z'_m] \in \mathbf{Z}_i$, given particle $\mathbf{s}_i^{(j)} = \left[r_i^{(j)}, r_i^{'(j)}\right]^T$, is obtained based on the joint probabilities of each state element, i.e. r and r' as

$$p\left(\mathbf{z}_{m,i}|\mathbf{x} = \mathbf{s}_i^{(j)}\right) = p_r(z_m|r_i^{(j)})p_r(z'_m|r_i^{'(j)}), \tag{3.7}$$

and $p_r(z|r)$ is computed using a Gaussian with mean r and standard deviation $\psi(r, z)$, defined shortly. This results in

$$p_r(z|r) = \frac{c * h(z)}{\sqrt{2\pi}\psi(r,z)} \exp\{-\frac{(z-r)^2}{2\psi(r,z)^2}\} \tag{3.8}$$

where c is a normalizing constant, and $h(z)$ is the normalized magnitude of the derivative at peak position z. For a set containing all choices of the first peaks, $\mathbf{Z}_1 \subseteq \mathbf{Z}^n$, the peak function $h(z_m)$ for each $z_m \in \mathbf{Z}_1$ is defined as

$$h(z_m) = \frac{|D_i(z_m)|}{\sum_{z_m \in \mathbf{Z}_1} |D_i(z_m)|}. \tag{3.9}$$

We have included $h(z)$ in our formulation to favour peaks with higher magnitude, hoping these peaks are located around the inner/outer boundaries. Alternatively, $h(z)$ can be defined as the normalized log of magnitude of derivative peaks. We have deployed the first formulation in our design considering the small range of magnitudes. Another function introduced in Equation 3.8 is $\psi(r, z)$, which provides a "range of effect" for each element of a particle, depending on particle's relative position to observation data and to other particles in the sample set. A large value of $\psi(r, z)$ is assigned if a particle's distance to its nearest peak is larger than the distance of other particles to their respective nearest observation peaks. Using this function, the particles that are farther away from the observation data are also assigned some weight, even though these weights will be small. Mathematically, in order to find the range of effect of r, the first element of particle $\mathbf{s}_i^{(j)} = \left[r_i^{(j)}, r_i^{\prime(j)}\right]^T$, where the set containing the first elements of observation data is represented by $\mathbf{Z}_1$, we define $\mathbf{q}(r)$ as

$$\begin{aligned} \mathbf{q}(r) &= [q_1, q_2, \cdots, q_N] \ \text{for}\, j = 1, \cdots, N \\ q_j &= \min\left\{v_{m,j} |\ v_{m,j} = |z_m - r^{(j)}| \ \text{ for each}\, z_m \in \mathbf{Z}_1\right\} \end{aligned} \tag{3.10}$$

Vector $\mathbf{q}(r)$ represents the distance between value r of all particles to their respective nearest measurement peak. Elements of this vector are linearly mapped to a constant sigma range $[\sigma_{min}, \sigma_{max}]$ to yield the ψ function as

$$\psi(r^{(j)}) = \frac{q_j - \min \mathbf{q}(r)}{\max \mathbf{q}(r) - \min \mathbf{q}(r)}(\sigma_{max} - \sigma_{min}) + \sigma_{min} \tag{3.11}$$

Weights derived based on the likelihood function of Equation 3.6 are normalized and assigned to particles to construct the new sample set representing the posterior distribution for this stage.

3.2.3 Resampling

A common problem with standard particle filter recursion is its degeneracy problem [37]. Without a resampling stage, weights from the previous step are carried to the next step, multiplied by the weights calculated in that iteration. This sometimes leads to the problem that after a few recursive steps, all but a few particles will have negligible weights. This is known as the degeneracy phenomenon, and it is remedied—at least to some extent— by adding a resampling step. Resampling eliminates particles with small weights, whereas particles with high weights may be selected multiple times, which will be diffused during the prediction stage. Since particles are resampled based on weights it makes sense to use cumulative distribution function (CDF) of weights to simplify the task of particle selection. Having sample set $\mathbf{SS_k} = \left\{ \mathbf{s}_k^{(j)}, w_k^{(j)} \right\}$, we can build cumulative weights as follow [16]:

$$\begin{aligned} c_k^{(0)} &= 0, \\ c_k^{(j)} &= c_k^{(j-1)} + w_k^{(j)} \text{ for } j = 1, \cdots, N \end{aligned} \tag{3.12}$$

Resampling is then performed in four steps:

- Draw random number u from uniform distribution: $u \sim \mathcal{U}[0, 1]$
- Find the smallest j for which $c_k^{(j)} \geq u$
- Set $\mathbf{s}'^{(n)}_{k+1} = \mathbf{s}_k^{(j)}$
- Assign uniform weights, $w_{k+1}^{(n)} = \frac{1}{N}$

In a noisy environment such as our MR images, where the observation data consist of multiple hyphotheses, the particles sometimes represent an incorrect density. If the degeneracy problem is not corrected, the faulty distribution is propagated to the next iteration, decreasing the chances of recovering from the error. We have experimented with the idea of enhanced resampling, which consists of manual insertion of particles at the observation data during the resampling phase. This method proved to be useful in some cases, where the density of particles around the true boundary was small. However, we should not forget the negative effect of this process, as the noisy observations dominate the distribution.

3.2.4 Line Convergence Criteria

Even though observation data of an individual line does not change from one iteration to the next, the random diffusion and resampling steps create a different set of particles at each iteration. Given that the particle filter is a heuristic approach, we hope that after a few iterations, the sample set will converge to a close approximation of the density. *Kullback-Leibler (KL) distance* or *asymmetric divergence* [39] gives a measure of the 'distance' or divergence between two distributions $p(x)$ and $\tilde{p}(x)$ as

$$KL = -\int p(x) \ln \frac{\tilde{p}(x)}{p(x)} \tag{3.13}$$

where $p(x)$ is assumed to be the true density. Even though this method provides a quantitative measure for comparing densities, there are two issues with applying it to our case. First, particle filters do not represent a continuous density function explicitly. The density can be computed based on the distribution of particles, but it is a slow process. The second problem is related to the fact that the KL-divergence considers $p(x)$ to be the true density, which does not hold true for our problem. In other words, if the posterior from the previous iteration $p(x)$ is noisy, the algorithm will converge if the current posterior $\tilde{p}(x)$ has the same noise pattern as the previous one. Therefore, we have proposed an alternative convergence criteria based on particle weights. At each iteration if a large percentage of particle weights fall below a certain weight threshold the algorithm converges. Mathematically put, the iteration stops if

$$\left| \left\{ w_i^{(j)} | w_i^{(j)} \leq 0.5 \times \max(w_i) \right\} \right| > 0.8N. \tag{3.14}$$

3.2.5 Line Propagation

After the convergence, we have a probability distribution for line i, represented by sample set $\mathbf{SS_i} = \left\{ \mathbf{s}_i^{(j)}, w_i^{(j)} \right\}$. This discrete distribution is stored before it is subjected to the resampling step. The small angular separation $\Delta\omega$ between adjacent lines enables us to use the posterior sample set from the previous line as the prior density of the current line. Therefore, our sample set is propagated to the next line in counter clockwise direction and the same iterative process is performed to converge to a posterior estimation for the next line and so on. This process continues until all lines have been visited once and we have stored $\mathbf{SS_k} = \left\{\mathbf{s}_k^{(j)}, w_k^{(j)}\right\}$ where $\mathbf{k} = (i, l, 1, 0)$ for each line i. Even though we obtain a probability distribution for each line during the primary contour derivation stage, we do not choose specific particles representing the boundaries. Instead the top A highest-weighting particles are saved to be used for the contour adjustment stage. Having higher weights increases the chances of getting selected during the resampling stage discussed in Section 3.2.3. The final boundary position determination based on sample sets is postponed to after the contour refinement step, discussed below.

3.3 Iterative Contour Refinement (ICR)

In the previous section we discussed different stages of a recursive probabilistic framework for estimation of density distributions for each radial line in the first frame.

In this section, we build upon our initial estimate to further refine probability distributions by utilizing state information from neighbouring lines. Figure 3.5 represents a flowchart, containing different steps of the ICR. Initially, we randomly choose a line i in frame t, and a traversal direction, CW or CCW. If it is the first iteration of the ICR, the sample set from the primary contour derivation stage is used as the prior for line i; otherwise, the posterior of the previous iteration is used. Each iteration of this stage involves traversing the radial lines and computing a revised probability distribution for each line, propagating density functions from one line to the next as described before. During this process, the same prediction and resampling steps, discussed in Sections 3.2.1 and 3.2.3 are applied. However, the observation stage is somewhat different as the likelihood

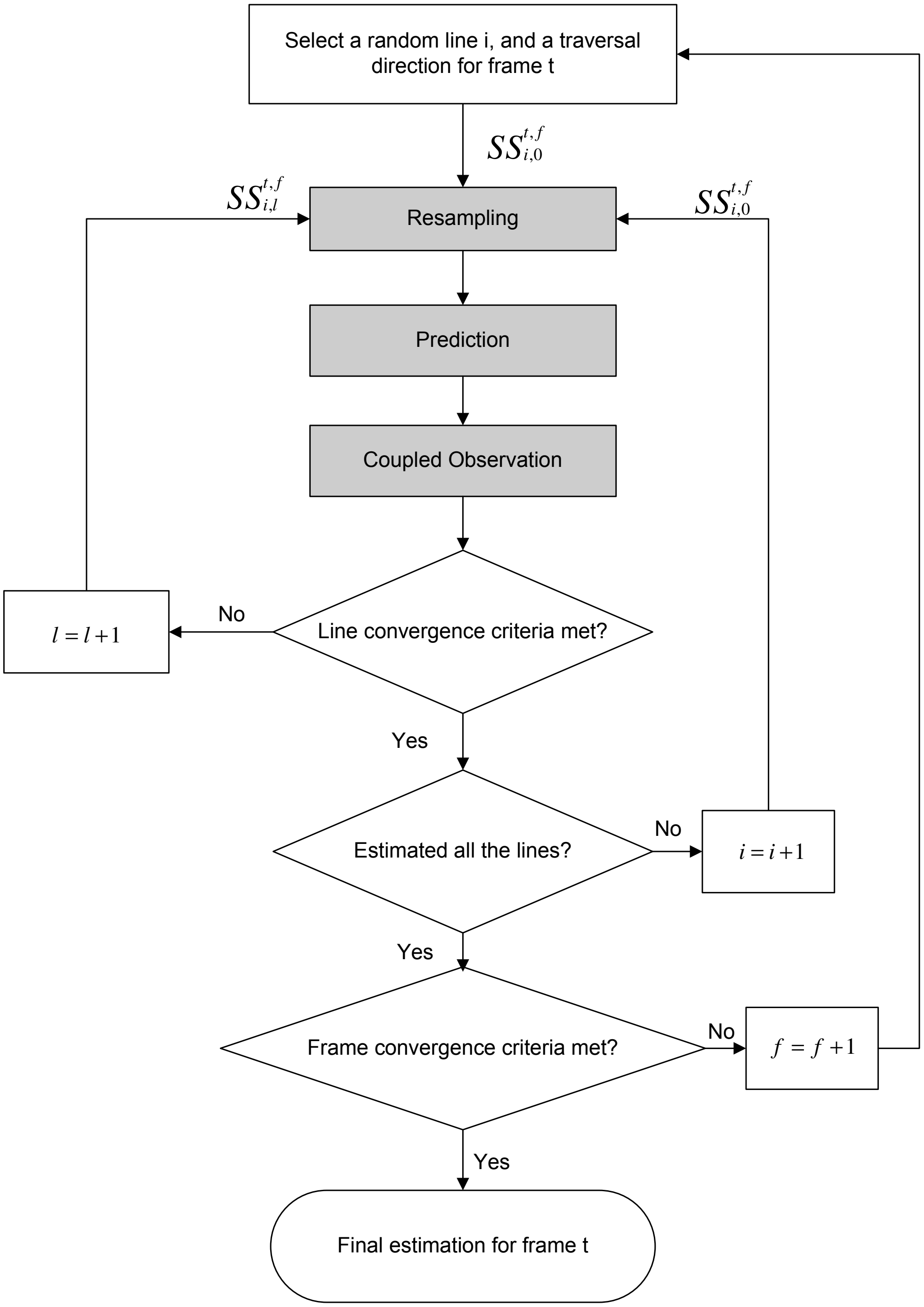


Figure 3.5: Iterative Contour Refinement Flowchart

function considers both measurements on the current line, i.e. derivative profile, and the top A states on the immediate neighbouring lines. Due to spatial coherence between neighbouring lines, we can impose continuity and smoothness constraints as previously reviewed in the active contour framework, Section 2.3.2. The next section provides a more in-depth description of our updated observation stage.

3.3.1 Coupled Observation Density

In the previous section we discussed the derivation of the likelihood function considering only the data measurements (i.e. intensity derivative peaks) for each line, resulting in updating weights based on $w_i^{(j)} = p(\mathbf{Z}_i|\mathbf{x} = \mathbf{s}_i^{(j)})$. In this section we introduce another measurement parameter $\mathbf{Y}_i$ for each line. Assuming that the likelihood probabilities for the two set of measurements $\mathbf{Z}_i$ and $\mathbf{Y}_i$ are independent, the likelihood function for particle $\mathbf{s}_i^{(j)}$ results in weight updates as

$$w_i^{(j)} = p(\mathbf{Z}_i|\mathbf{x} = \mathbf{s}_i^{(j)})p(\mathbf{Y}_i|\mathbf{x} = \mathbf{s}_i^{(j)}) \tag{3.15}$$

where $p(\mathbf{Z}_i|\mathbf{x})$ is derived the same way as Equation 3.6, but how do we define $\mathbf{Y}_i$ for each line? Recall from the previous section we carry along the top A (defined as 4 in this case) estimates, resulting in

$$\mathbf{E}_i = \Big\{\mathbf{e}_i^{(a)} = \begin{bmatrix} r_i^{(a)} \\ r_i^{'(a)} \end{bmatrix} \text{ for } \quad a = 1, \cdots, A\Big\}. \tag{3.16}$$

We can construct the observation data set $\mathbf{Y}_i$ by forming quadruple vectors, each vector defined based on a unique permutation of state elements in the top A candidates in the previous and next lines. In other words

$$\mathbf{Y}_i = \left\{(r_{i-1}^{(a1)}, r_{i-1}^{'(a2)}, r_{i+1}^{(a3)}, r_{i+1}^{'(a4)})\right\} \text{ for different permutations of } \quad a1,\ a2,\ a3,\ a4 = 1, \cdots, A. \tag{3.17}$$

The total number of observations is proportional to the number of selected particles, and is given by $M_y = A^4$, considering the four elements of the quadruples. The likelihood of $\mathbf{Y}$ observations given sample $\mathbf{x} = \mathbf{s}_i^{(j)}$ is given by

$$p(\mathbf{Y}_i | \mathbf{x} = \mathbf{s}_i^{(j)}) = \sum_{b=1}^{M_y} p(\mathbf{y}_b | \mathbf{x}). \tag{3.18}$$

Each triple of particles—two on the adjacent lines and particle j on the current line—are evaluated with regards to two criteria, the continuity constraint and the smoothness constraint. The continuity constraint poses limitations on variations of state positions r and r' from one line to the next, whereas the smoothness constraint ensures that each of the inner/outer boundaries are relatively smooth, without rough edges or corners. The probability distribution of each measurement $\mathbf{y}_i = (r_{i-1}^{(a1)}, r_{i-1}^{'(a2)}, r_{i+1}^{(a3)}, r_{i+1}^{'(a4)})$, is taken as

$$p(\mathbf{y}_i \big| \mathbf{x} = \mathbf{s}_i^{(j)}) = p_c(\mathbf{y}_i \big| \mathbf{s}_i^{(j)}) p_s(\mathbf{y}_i \big| \mathbf{s}_i^{(j)}). \tag{3.19}$$

Recall that the continuity constraint controls deviations of the current estimate with neighbouring lines. The continuity function C for r (or r') values on three consecutive lines $i-1, i, i+1$ is defined as:

$$C(r_{i-1}, r_i, r_{i+1}) = \frac{1}{2}\big[(r_i - r_{i-1})^2 + (r_i - r_{i+1})^2.\big] \tag{3.20}$$

Since the continuity constraint $C(\cdot)$ is always positive, a half Gaussian distribution (note the normalizing factor of two in the numerator) is used to derive the continuity probability

p_c for each contour

$$\begin{aligned} p_c(\mathbf{y}_i|\mathbf{s}_i^{(j)}) &= p_c(r_{i-1}^{(a)}, r_{i+1}^{(a)}|r_i^{(j)})p_c(r_{i-1}^{'(a)}, r_{i+1}^{'(a)}|r_i^{'(j)}) \\ p_c(r_{i-1}^{(a)}, r_{i+1}^{(a)}|r_i^{(j)}) &= \frac{2}{\sqrt{2\pi}\sigma_c}\exp\{-\frac{C(r_{i-1}^{(a)}, r_{i+1}^{(a)}, r_i^{(j)})}{2\sigma_c^2}\} \\ p_c(r_{i-1}^{'(a)}, r_{i+1}^{'(a)}|r_i^{'(j)}) &= \frac{2}{\sqrt{2\pi}\sigma_c}\exp\{-\frac{C(r_{i-1}^{'(a)}, r_{i+1}^{'(a)}, r_i^{'(j)})}{2\sigma_c^2}\} \end{aligned} \tag{3.21}$$

Equation 3.21 indicates that the chosen probability is a Gaussian distribution in $C(\cdot)$ with zero mean and standard deviation σ_c. The density function favours smaller values of the continuity constraint, defined above. The effect of the continuity constraint on the overall likelihood calculation is controlled by the user-defined parameter σ_c. While large values of σ_c allow more variations in state positions *vs.* neighbouring lines, smaller values limit particle positions to be close to the states of adjacent lines.

In addition to the continuity constraint defined in Equation 3.21, we also define the smoothness probability distribution p_s as

$$\begin{aligned} p_s(\mathbf{y}_i|\mathbf{s}_i^{(j)}) &= p_s(r_{i-1}^{(a)}, r_{i+1}^{(a)}|r_i^{(j)})p_s(r_{i-1}^{'(a)}, r_{i+1}^{'(a)}|r_i^{'(j)}) \\ p_s(r_{i-1}^{(a)}, r_{i+1}^{(a)}|r_i^{(j)}) &= \frac{1}{\sqrt{2\pi}\sigma_\alpha}\exp\{-\frac{(\alpha_{in}-\alpha_m)^2}{2\sigma_\alpha^2}\} \\ p_s(r_{i-1}^{'(a)}, r_{i+1}^{'(a)}|r_i^{'(j)}) &= \frac{1}{\sqrt{2\pi}\sigma_\alpha}\exp\{-\frac{(\alpha_{out}-\alpha_m)^2}{2\sigma_\alpha^2}\} \end{aligned} \tag{3.22}$$

where α_{in} and α_{out} are illustrated in Figure 3.6. The mean and standard deviation of this Gaussian distribution, i.e. α_m and σ_α are parameters that are derived based on empirical results and analysis of hand-segmented data. In general, left ventricle inner and outer contours present smoothness angles of around $\alpha_m = 170°$. The standard variation, $\sigma\alpha$, is defined as 50° to allow for various inner contour smoothness angles. The effects of varying $\sigma\alpha$ are discussed in more details in Chapter 4.

For $\Delta\theta = 15°$ and $L = 24$, the expected smoothness angle for a circle is equal to $\alpha_m = 165°$. This number will increase to $\alpha_m = 170°$ for $\Delta\theta = 10°$.

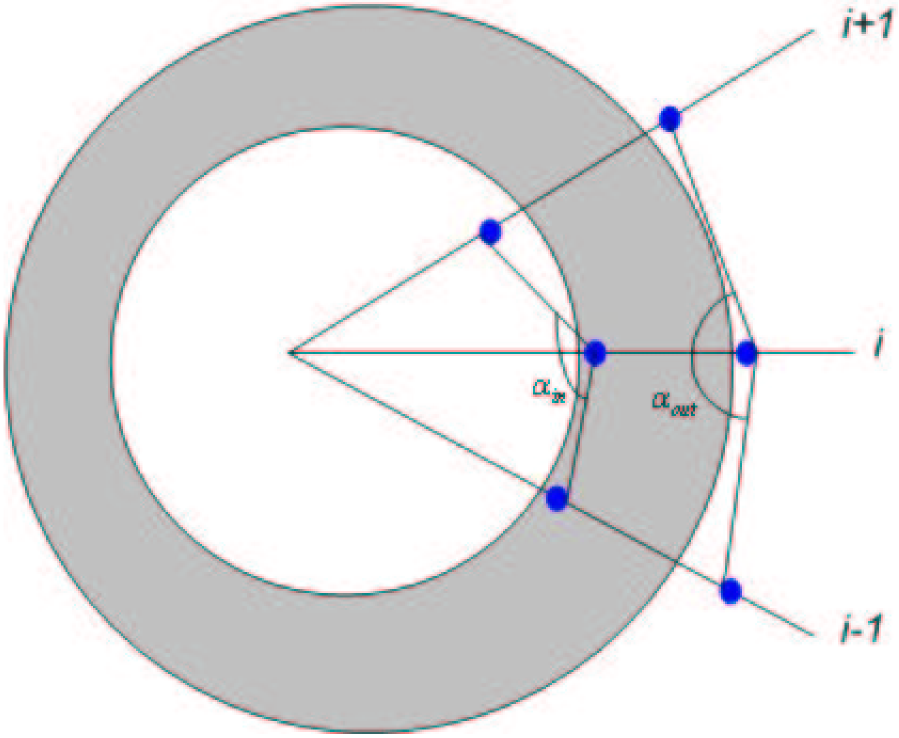


Figure 3.6: Smoothness Angles

3.3.2 State Estimation

Based on the iterative process described in Section 3.3, all the radial lines are visited and state density for each line is determined. The density estimate for the first line of the current frame is used as the prior for PCD stage of the subsequent frame. The same refinement step is performed, resulting in derivation of contour positions for each frame in a dataset. However, we still need a way to estimate the position of boundaries based on these distributions. The weighted average of top A particles after each iteration is used to represent the final state for each line. The frame iteration converges if none of the final estimates moves more than a certain threshold from one iteration to the next. This threshold is defined as 4 pixels in our design. Spline fitting is performed to connect the points around the inner and outer borders, yielding a complete contour derivation for all the frames.

Chapter 4

Results

We provided a detailed description of our overall design and its various components in Chapter 3. In this chapter we demonstrate some of the results obtained by running our algorithm on short-axis cine MRI data sets. Each complete data set contains 12 slices of short-axis MR images, each slice captured at 20 phases or frames. Unless otherwise specified, the presented results are generated using the second slice (Z_2) of the sample data set (C_1). The original size of each 2D scan is 256×256 pixels; however, zooming in the region of interest yields image sizes of roughly 100×100 pixels. Processing each frame with around 10 iterations takes approximately 100s on Pentium V, 2.67GHz, running Linux. All simulations are in Matlab, and an optimized performance can be achieved by implementing the algorithm in C++. Qualitative results are mainly presented by visualization of a 2D scan, overlaid by the top three candidates for the boundary positions on each line. Section 4.1 illustrates boundary estimations using only the primary contour derivation method. The effect of adding the continuity and the smoothness constraints are demonstrated in Section 4.2. Finally, quantitave measures, comparing our method with the hand-segmented data are presented in Section 4.3.

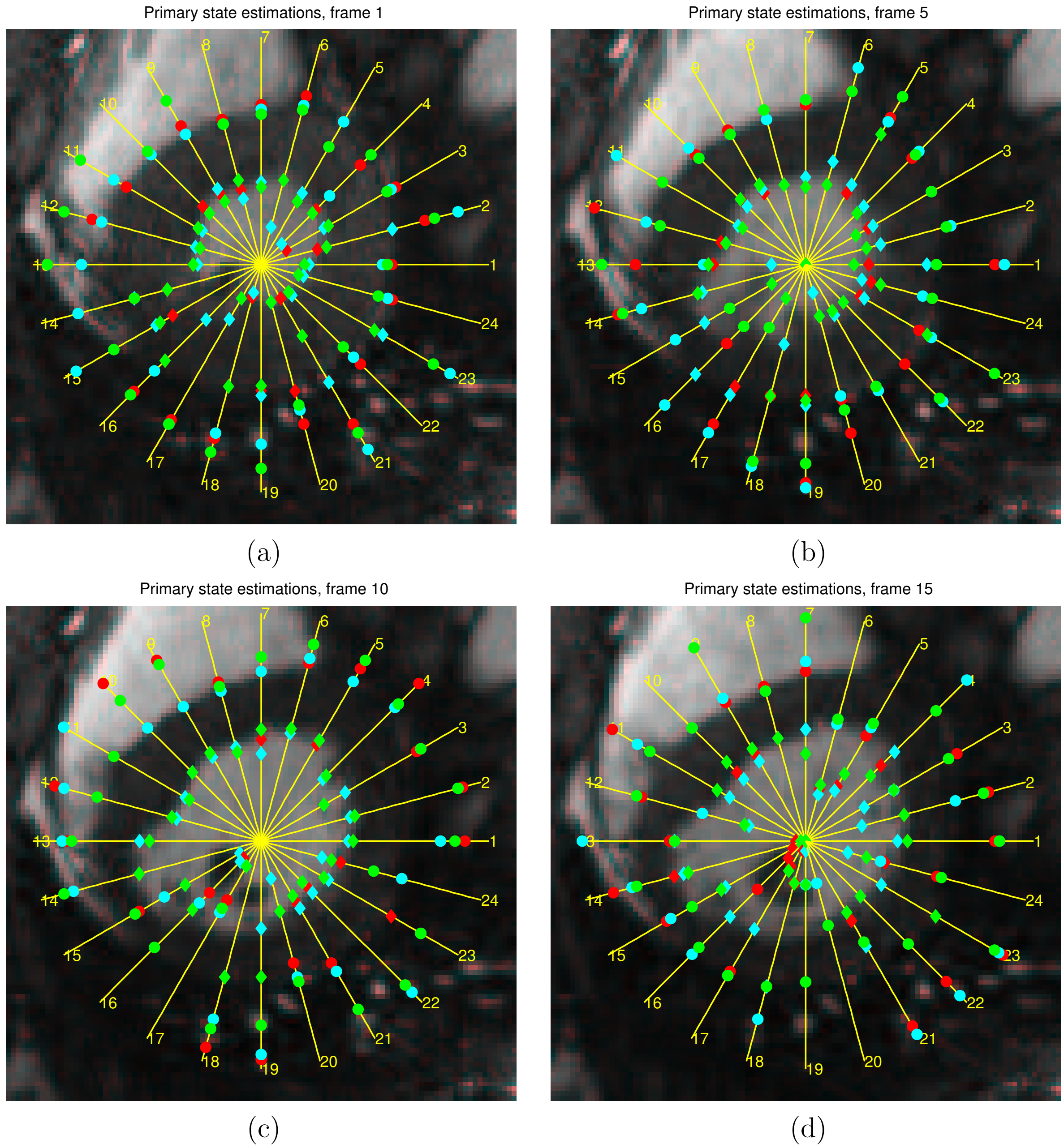
4.1 Primary Contour Derivation Results

The initial boundaries are calculated using the primary contour derivation algorithm discussed in Section 3.2. However, as we mentioned before, the output of this stage is a set of particles that carry the highest weights for each line. Figure 4.1 depicts some of these results for various frames of our selected data set. In each figure, the top-three ranking particles are displayed with red, cyan and green colours respectively.

We also make a distinction between the estimates for the inner and outer boundaries by using diamond markers for the inner boundary estimates, and circle markers representing the outer boundary estimates. Angular sampling of $\Delta\omega = 15^{\circ}$ and a total number of samples $N = 50$ is used to generate these results. As shown, at least one of the particles is a true representation of the boundary in most of the lines. The papillary muscles (the two black areas in the ventricular cavity) have similar intensity values as the myocardium and differentiating them from the LV wall poses a great challenge to all vision-based algorithms. The effect of these muscles is best seen around lines 16–19, where one or all of the particles are displaced, as shown in Figure 4.1.c.

The same frames are tested using an enhanced resampling stage, and the results are presented in Figure 4.2. Recall from Section 3.2.3 that the original resampling stage is enhanced by inserting a particle at each one of the observation locations manually.

Qualitative comparison of the frames in both Figure 4.1 (without adjustments) and 4.2 (with adjustments) reveals that while enhanced resampling improves the estimates in some cases, such as lines 20–23 in frame 15, it introduces noisy estimates in other cases, such as lines 16–19 of the same frame. Considering our data, it is likely to have spurious observation data due to noise or image artifacts. Therefore, we do not believe that introducing the enhanced resampling will be advantageous to us. The rest of the results presented in this chapter, are derived based on the original resampling rather than the enhanced version of it.

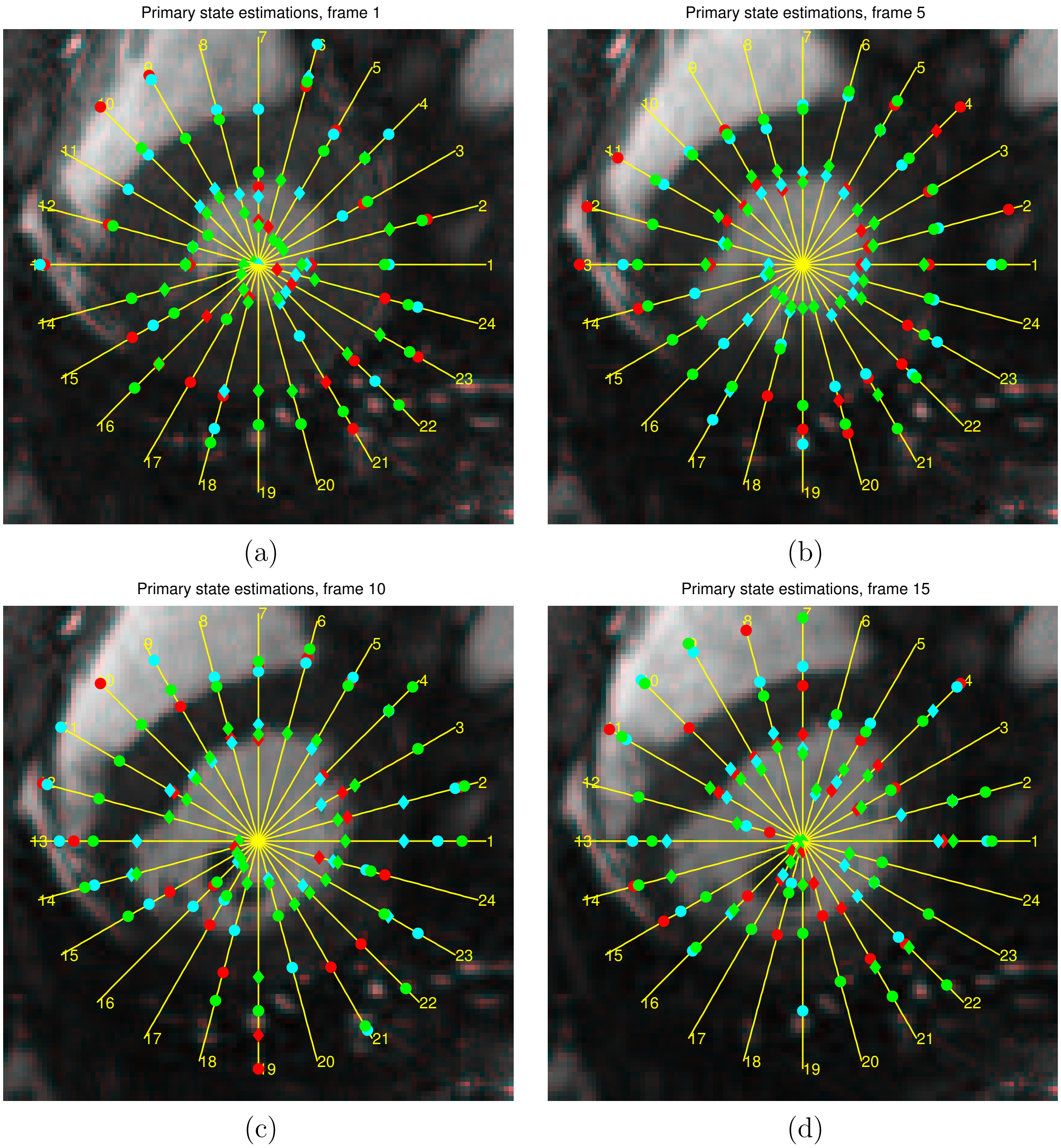

Primary state estimations, frame 1
Primary state estimations, frame 5
Primary state estimations, frame 10
Primary state estimations, frame 15
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

(a) (b)

(c) (d)

Figure 4.1: Primary contour derivation results using $N = 50$ particles, for (a) frame 1, (b) frame 5, (c) frame 10, and (d) frame 15, with high to low weights shown in red, cyan and green respectively

Figure 4.3 presents the same frames using $N = 200$ particles. Comparison of Figure 4.1 using $N = 50$ particles, and Figure 4.3, using $N = 200$, indicates that we do not gain significant accuracy by increasing the number of particles in this case.


Primary state estimations, frame 1

(a)

Primary state estimations, frame 5

(b)

Primary state estimations, frame 10

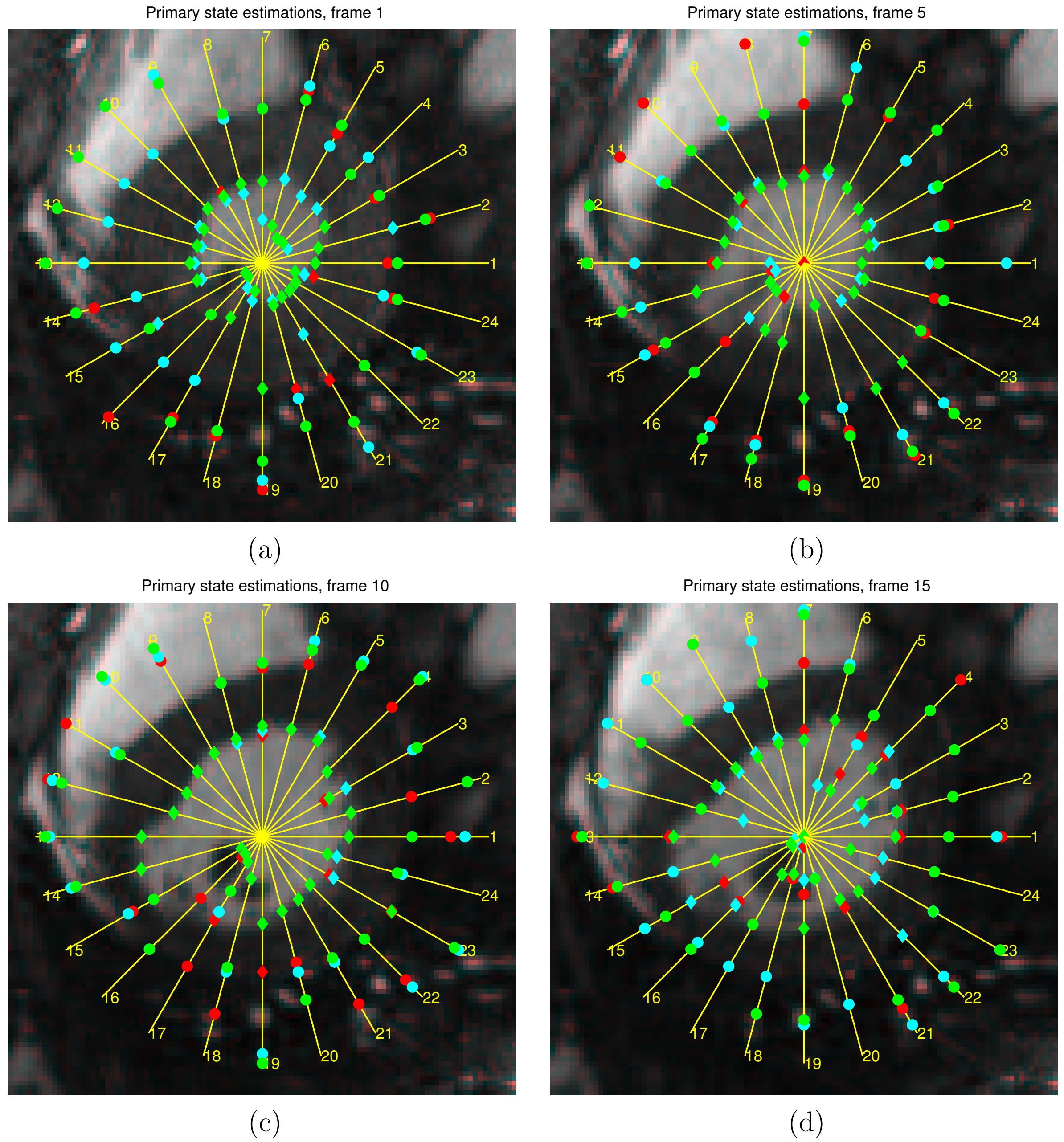
(c)

Primary state estimations, frame 15

(d)

Figure 4.2: Primary contour derivation results using enhanced resampling for (a) frame 1, (b) frame 5, (c) frame 10, and (d) frame 15, with high to low weights shown in red, cyan and green respectively

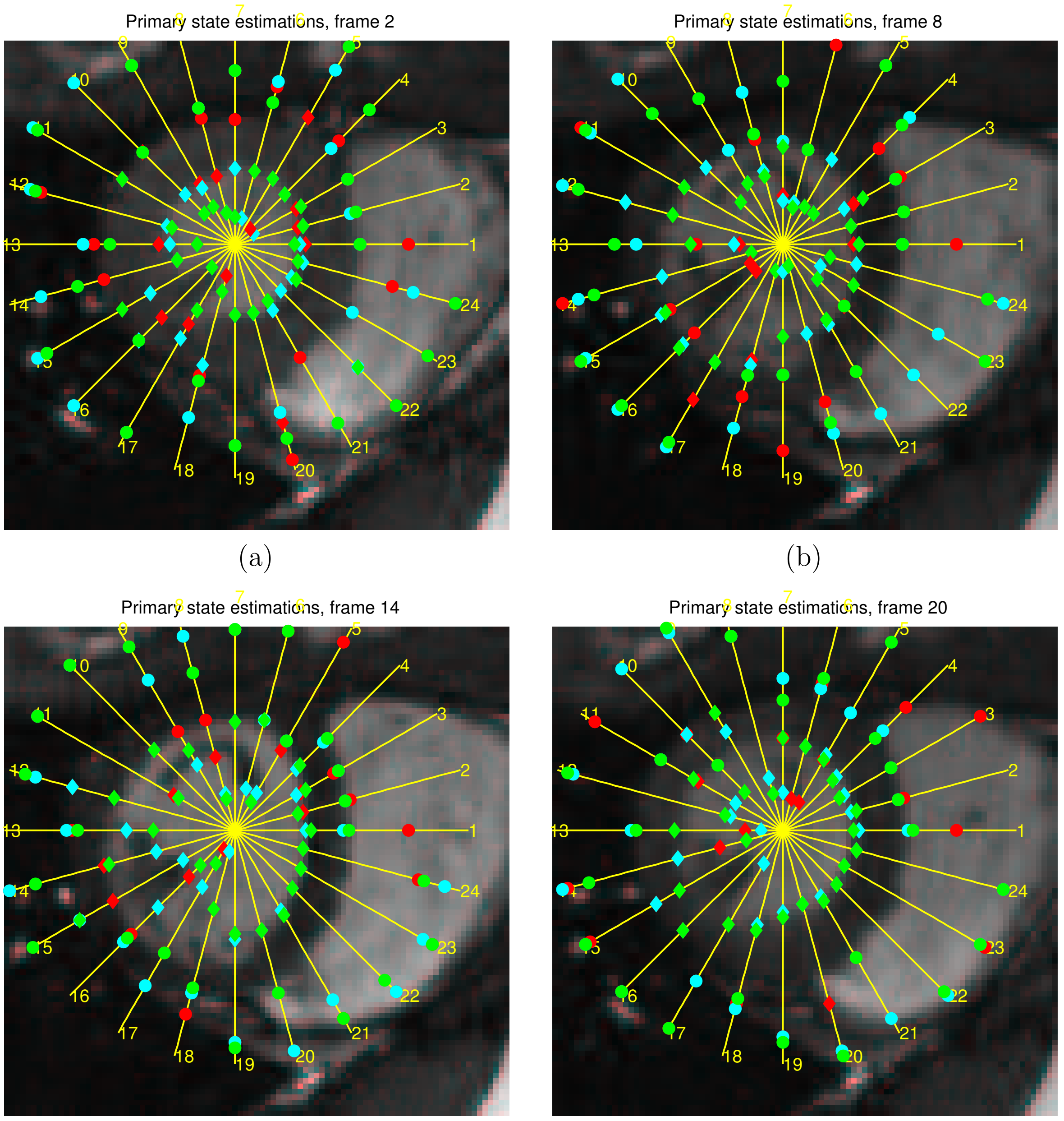
For the sake of comparison, we have tested our algorithm on the third slice (Z_3) of another data set, which we refer to as C_2. The lower range of intensities and more motion artifacts in the images make the segmentation process more challenging and less accurate as depicted in Figure 4.4.


Primary state estimations, frame 1
Primary state estimations, frame 5
Primary state estimations, frame 10
Primary state estimations, frame 15
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

(a) (b) (c) (d)

Figure 4.3: Primary contour derivation results using $N = 200$ particles for (a) frame 1, (b) frame 5, (c) frame 10, and (d) frame 15, with high to low weights shown in red, cyan and green respectively

As shown in Figure 4.4.c, most of the estimates for the outer boundary are displaced, due to small intensity variations at the epicardial borders. Even the estimates for lines 20–24 are erroneous, despite having high-gradient edges at these lines. This can be


Primary state estimations, frame 2
Primary state estimations, frame 8
Primary state estimations, frame 14
Primary state estimations, frame 20
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

(a)

(b)

(c)

(d)

Figure 4.4: Primary contour derivation results for C_2, using $N = 100$ particles for (a) frame 2, (b) frame 8, (c) frame 14, and (d) frame 20, with high to low weights shown in red, cyan and green respectively

associated with having small σ_r in the prediction stage, discussed in Section 3.2.1, which makes it hard to recover from incorrect estimates.

4.2 Iterative Contour Refinement Results

In this section, we further refine the position of particles on each line by coupling them to particles on adjacent lines. The top $A = 4$ particles on neighbouring lines provide additional constraints that are employed in better representation of the probability density of the current line. We only demonstrate the top three particles, although the data are calculated based on $A = 4$. We first discuss the effect of adding each of one the continuity and smoothness constraints—as described in Section 3.3.1—exclusively. Finally, we present the results for an integrated system including both measures. All the results of this section are obtained using the slice Z_2 of C_1, with $N = 60$.

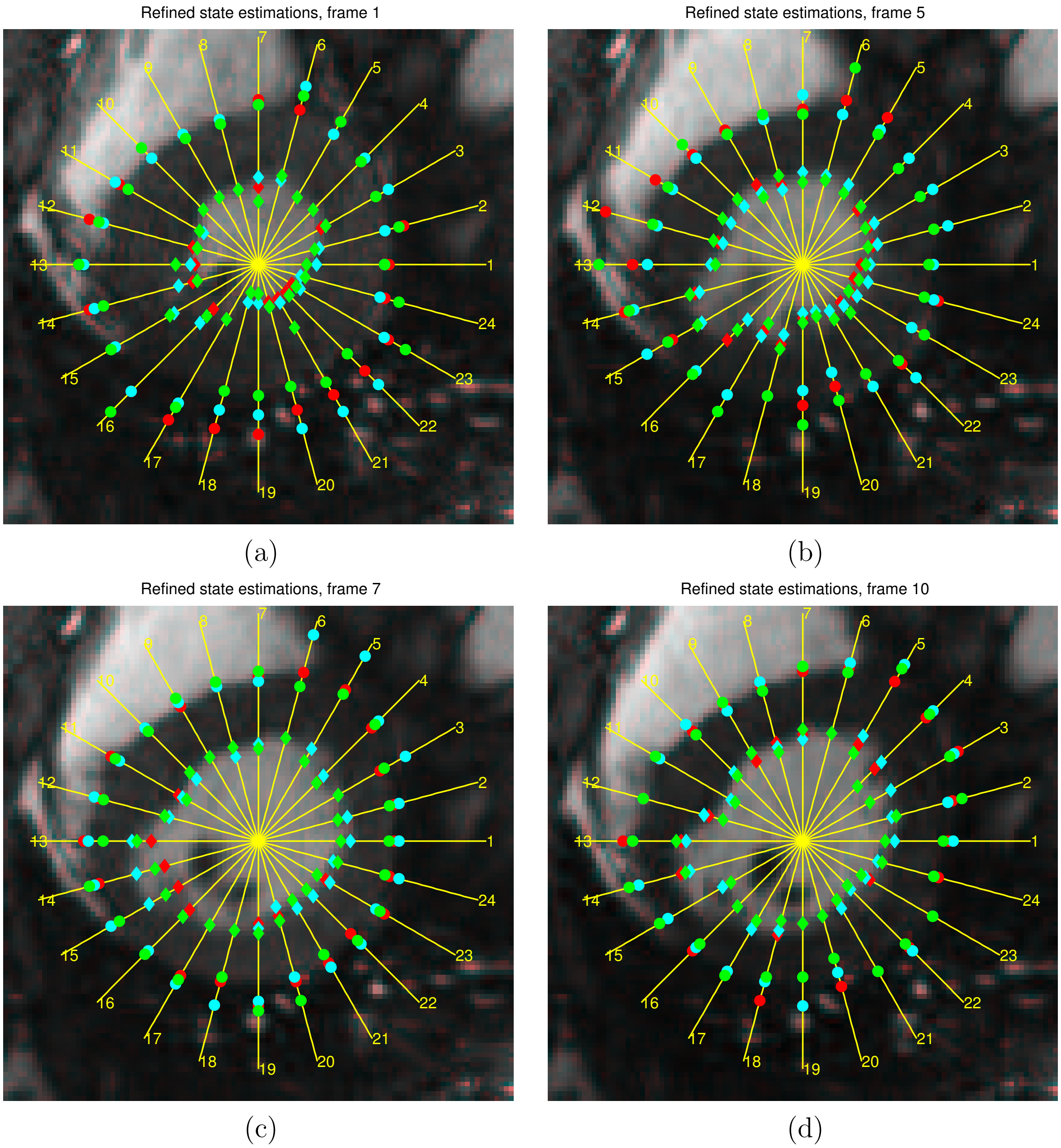
4.2.1 Continuity Constraint

In this section we experiment with including the continuity constraint in our likelihood function. Figure 4.5 depicts the results of including the continuity constraint in our formulation and estimating the boundary positions iteratively. Qualitative comparison of estimates in frame 1 after the PCD step, Figure 4.1.a, and after the ICR step, Figure 4.5 demonstrates how particles are converged towards the true boundary positions using the continuity constraint.

Figure 4.6 depicts the results for frame 10, using two different values of σ_c. Larger values of σ_c allow the particles to move more independently from their neighbouring particles; whereas the particles are strongly coupled when a small continuity variance is used.

4.2.2 Smoothness Constraint

This section demonstrates the effect of adding the smoothness constraint exclusively. Different values of σ_α are tested for the iterative contour refinement step, and the results are depicted in Figure 4.7.


Refined state estimations, frame 1
Refined state estimations, frame 5
Refined state estimations, frame 7
Refined state estimations, frame 10
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

(a)

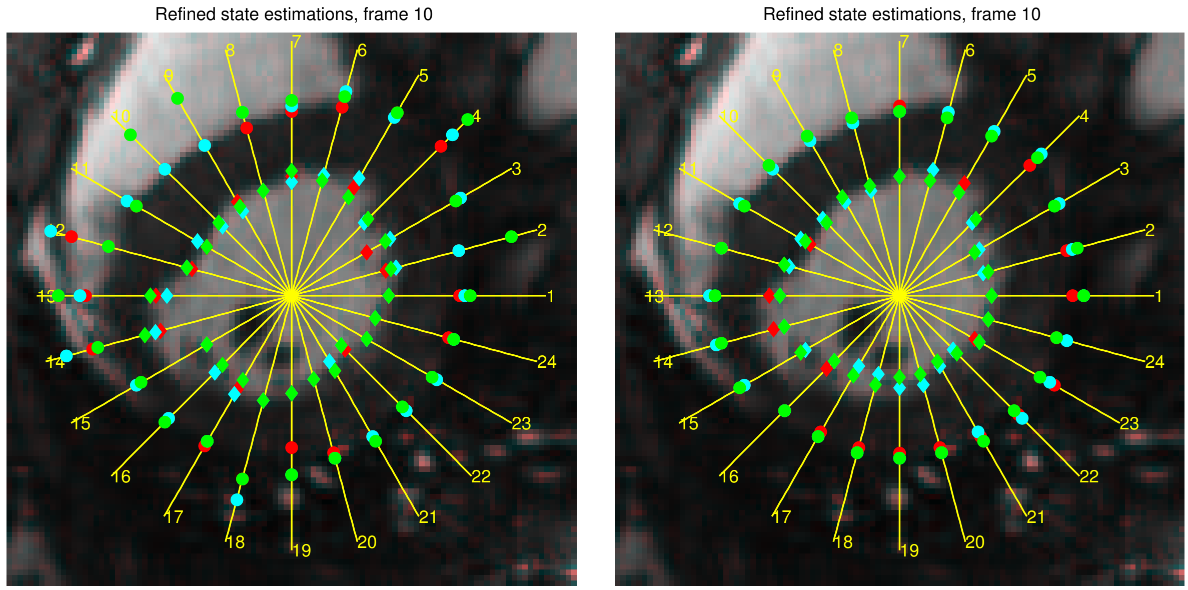
(b)

(c)

(d)

Figure 4.5: Iterative contour refinement results, using $\sigma_c = 7$, for (a) frame 1, (b) frame 5, (c) frame 7, and (d) frame 10

While smaller values of σ_α have an oversmoothing effect on the boundaries, larger values, as depicted in Figure 4.7.b give the contour the flexibility to latch on to intensity derivative peaks. The oversmoothing effect is evident around lines 13–18 in all the frames in Figure 4.8, when using $\sigma_\alpha = 30°$ to derive the boundary estimates. In order to improve the elasticity of our contours, we have chosen $\sigma_\alpha = 50°$ as our default.


Refined state estimations, frame 10
Refined state estimations, frame 10
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

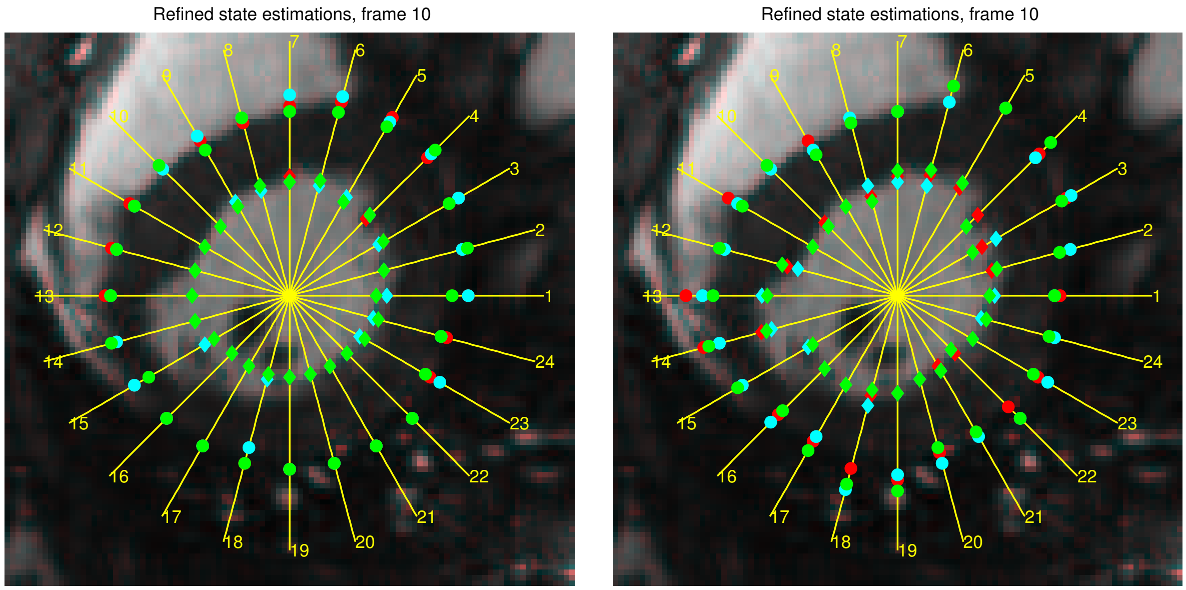
(a) (b)

Figure 4.6: Iterative contour refinement results for frame 10, using continuity constraint with (a) $\sigma_c = 10$, (b) $\sigma_c = 2$

4.2.3 Combined Refinement

At this stage, we use the combination of the above-mentioned measures, i.e. continuity and smoothness, in our formulation of the likelihood function. In order to make the comparison easier, we use the same frames that were used in Figure 4.1. As shown in Figure 4.9, the oversmoothing effect, although still present, is less significant than the previous case when only the smoothness constraint was used.

Different combination of σ_c and σ_α are used in our experiments with iterative contour refinement. Figure 4.10 demonstrates boundary estimations in frame 10, using various constraint values.

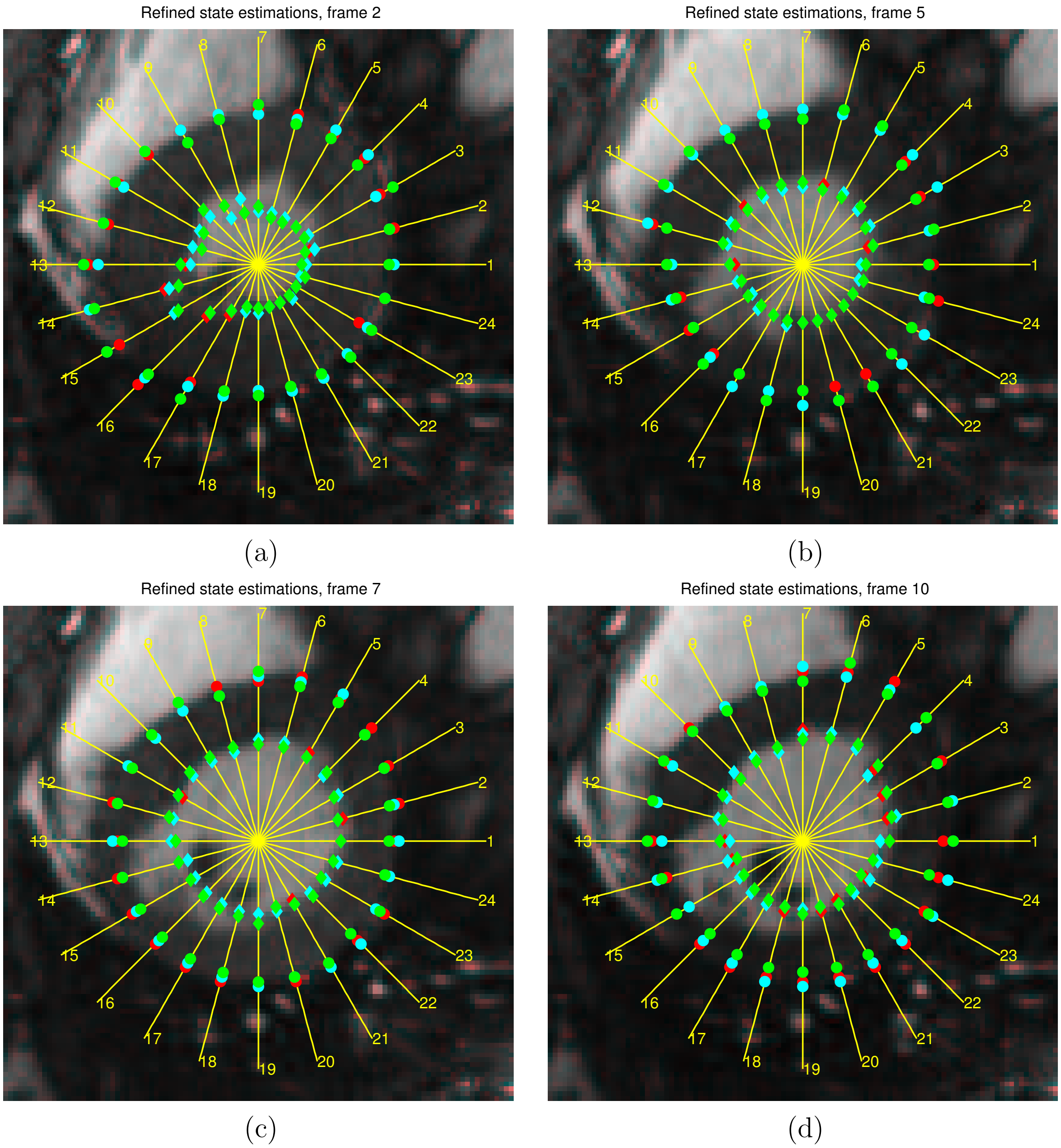

Refined state estimations, frame 10
Refined state estimations, frame 10
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

(a) (b)

Figure 4.7: Iterative contour refinement results for frame 10, using smoothness constraint with (a) $\sigma_\alpha = 20°$, (b) $\sigma_\alpha = 50°$

4.3 Quantitative Comparison with Hand-Segmented Data

Unfortunately, there are not any standard data sets which can be used to provide comparative analysis of our approach versus other methods described in Section 2.3. Therefore, we limit ourselves to comparing the accuracy of our method versus the hand-segmented data—provided by Ontario Consortium for Cardiac Imaging—for a sample slice. These data include a set of points around the boundary. The hand-segmented data are interpolated, specifying the boundary positions on each line. The error or distance is simply defined as the absolute value of the difference between the true and the estimated boundary positions for each contour. We have demonstrated the error values for a few sample frames, after the ICR step, in Figure 4.11. The spikes at line 5 in both frames of Figure 4.11, are associated with the fact that the epicardial border at this particular line is so low-contrast that no observations are found at this boundary location.


Refined state estimations, frame 2

(a)

Refined state estimations, frame 5

(b)

Refined state estimations, frame 7

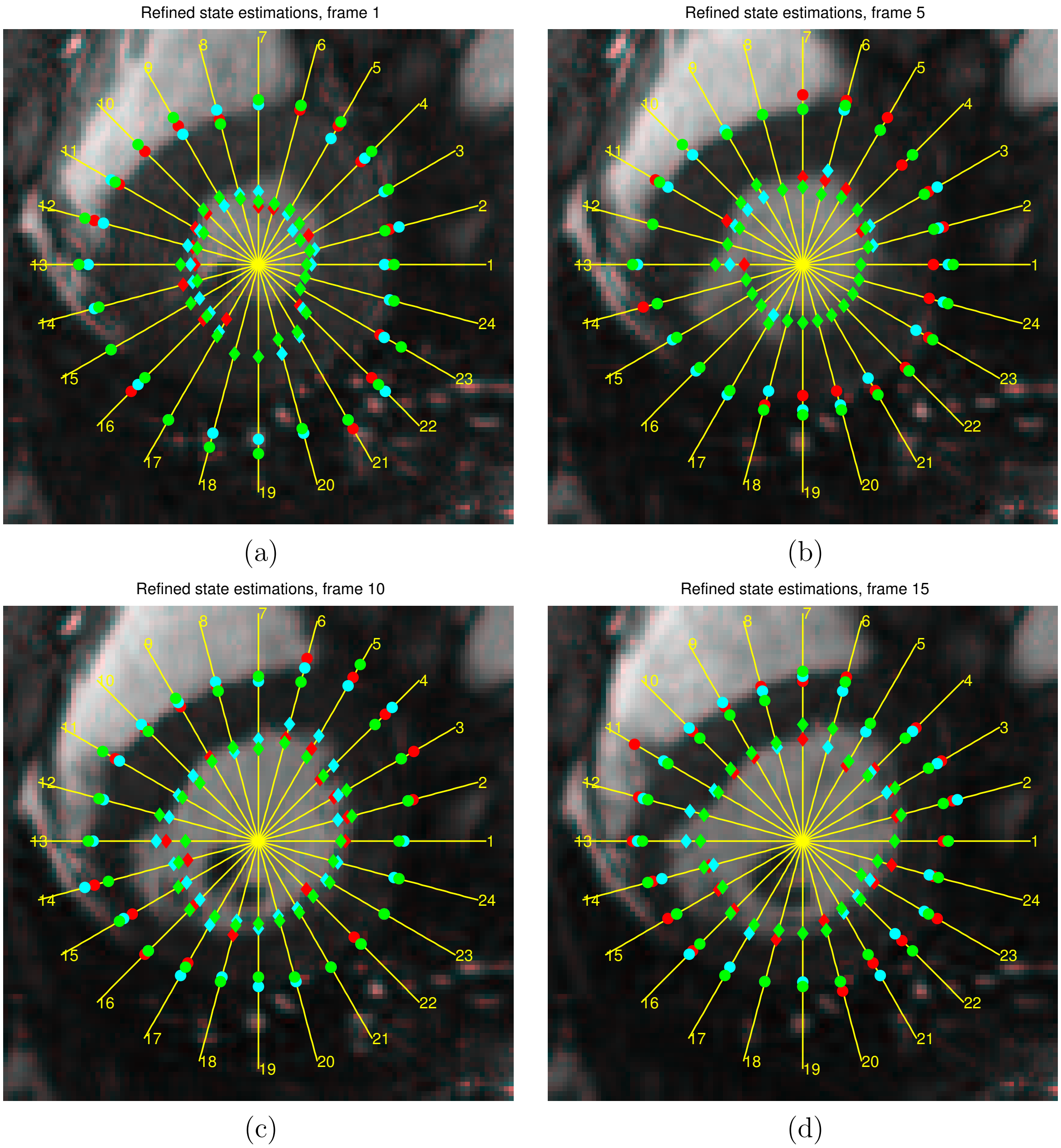
(c)

Refined state estimations, frame 10

(d)

Figure 4.8: Iterative contour refinement results, using smoothness constraint with $\sigma_\alpha = 30°$ for (a) frame 2, (b) frame 5, (c) frame 7, and (d) frame 10

The boundary positions for all 20 frames in this slice Z_2, are estimated based on our algorithm and compared with the hand-segmented counterpart in terms of root mean squared error (RMSE). The RMSE after the ICR step, is presented both per frame and per line as illustrated in Figure 4.12.


Refined state estimations, frame 1

(a)

Refined state estimations, frame 5

(b)

Refined state estimations, frame 10

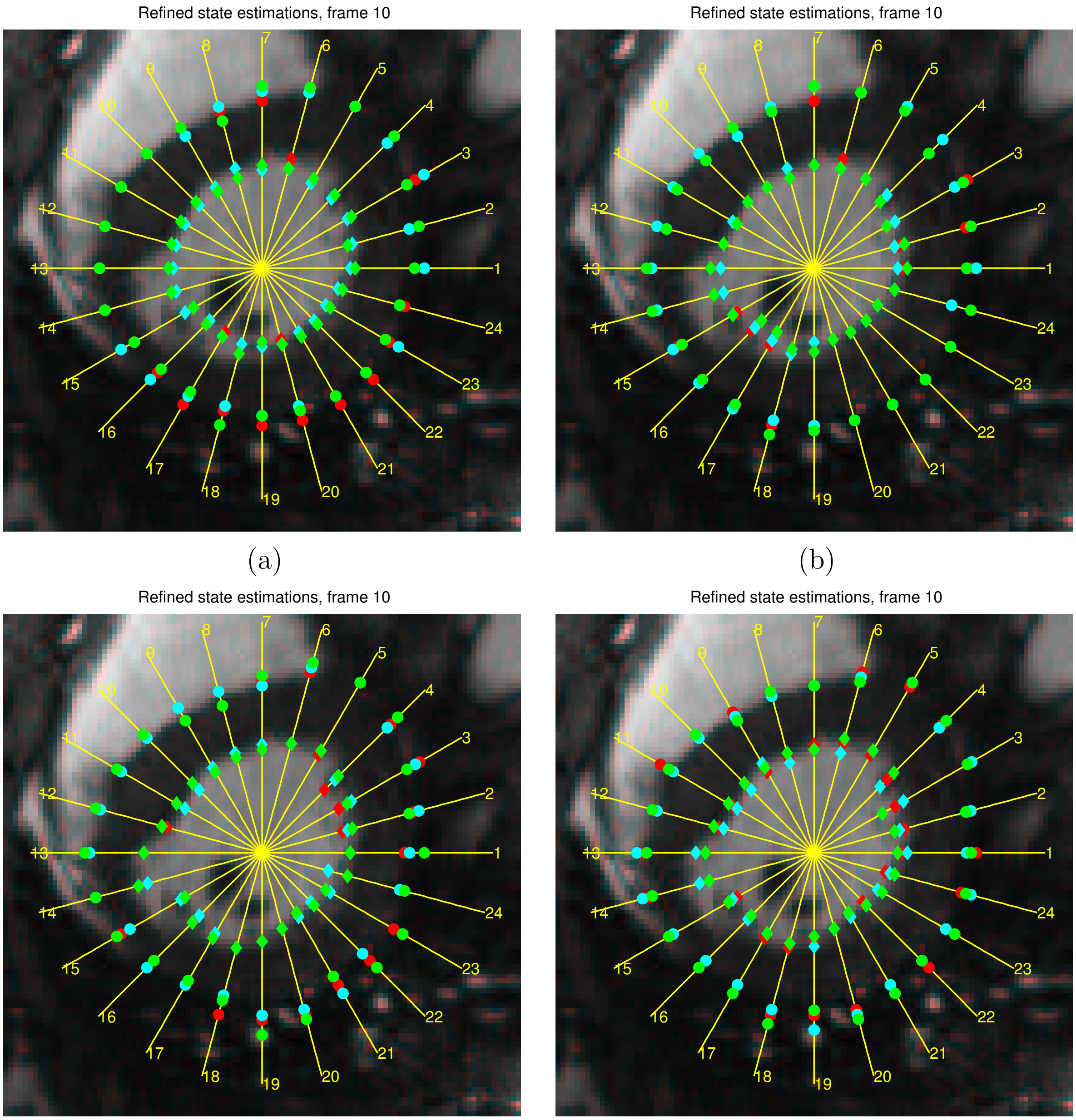
(c)

Refined state estimations, frame 15

(d)

Figure 4.9: Iterative contour refinement results, using $\sigma_\alpha = 50°$, and $\sigma_c = 10$, for (a) frame 1, (b) frame 5, (c) frame 10, and (d) frame 15

The same results for the primary contour derivation stage are presented in Figure 4.13. As it is evident from this figure, the erros in estimating inner and outer contours are strongly correlated, which is a direct result of the coupled estimation. The errors from one boundary normally affect the other as well and we can identify this as one of the problems of our method. Based on these values, more than 90% of frames have $RMSE \leq 5$ after



(a)

(b)

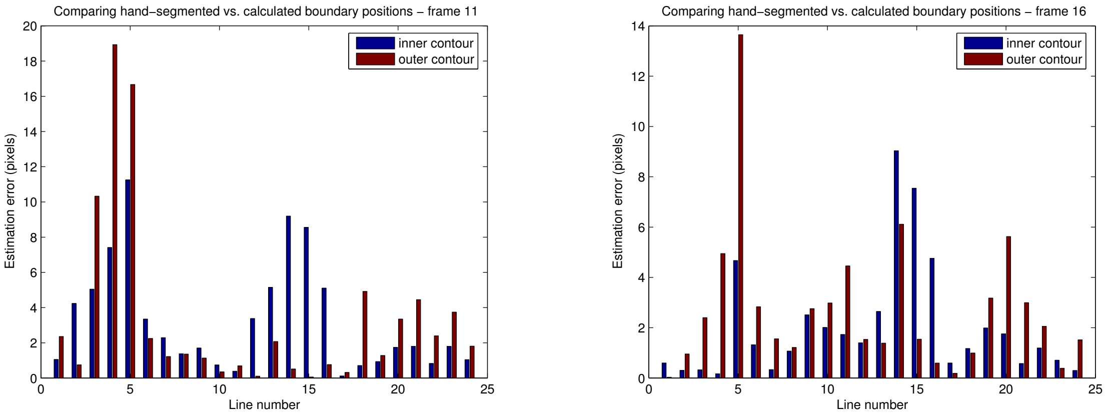
(c)

(d)

Figure 4.10: Iterative contour refinement results for frame 10, using (a) $\sigma_\alpha = 20°$, $\sigma_c = 10$, (b) $\sigma_\alpha = 20°$, $\sigma_c = 5$, (c) $\sigma_\alpha = 50°$, $\sigma_c = 10$, (d) $\sigma_\alpha = 50°$, $\sigma_c = 5$

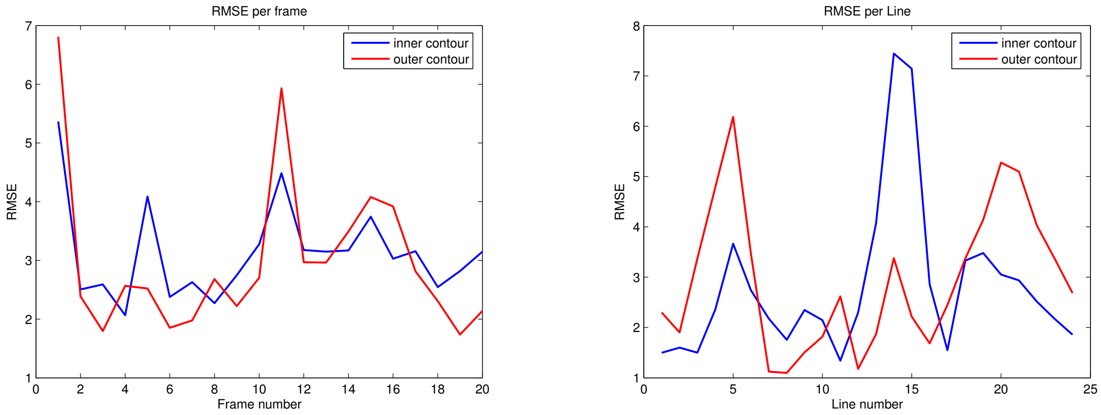
the ICR. Meanwhile, more than 80% of all the lines have $RMSE \leq 5$. Considering Figure 4.12, lines 5 and 15 have the most effect on the RMSE, one due to low-contrast boundaries and the other due to the papillary muscles as was discussed before.

Combining the estimations from all the lines and frames, we represent the occurence frequency of absolute errors for inner and outer boundaries using histograms. These



(a) (b)

Figure 4.11: ICR - Estimation errors for all the lines in frame (a) frame 11, (b) frame 16.

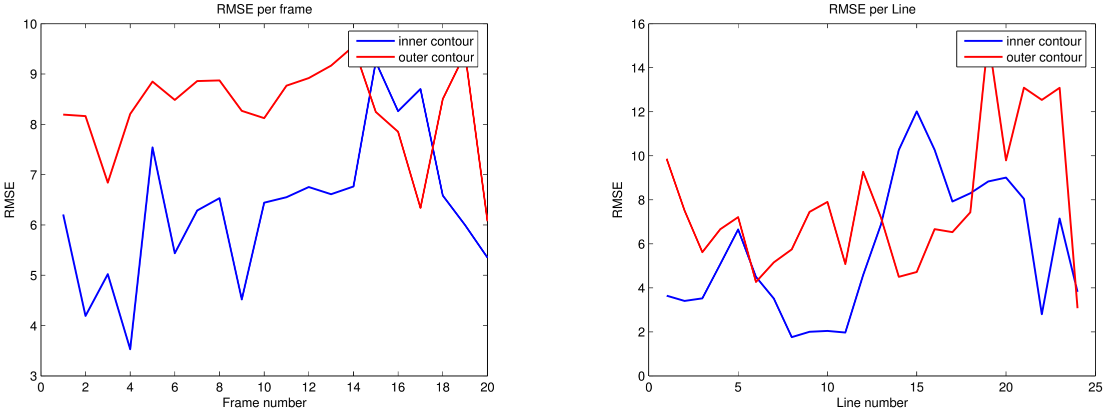


(a) (b)

Figure 4.12: ICR - Root mean squared error (RMSE) of boundary estimation (a) for all the frames (b) for all the lines

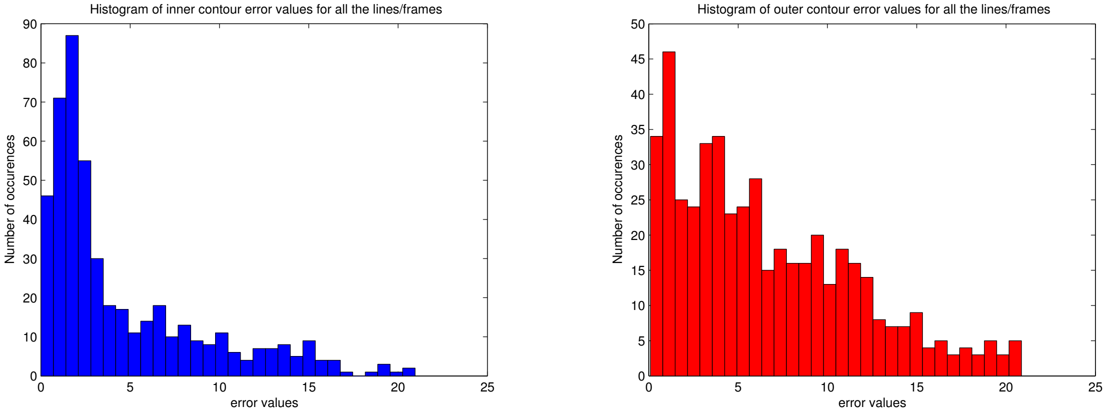
histograms are depicted in Figures 4.14, and 4.15 for PCD and ICR steps, respectively. Based on these data, more than 70% of all estimates fall within 2.5 pixels distance of the groundtruth data.

We use spline interpolation to derive the final contour from the estimated boundary positions. Figure 4.16 compares the final contour derived using our method versus the hand-segmented contours. As it is evident from the figure, there is a good correlation between the two methods, except for some oversmoothing observed in Figure 4.16.c.



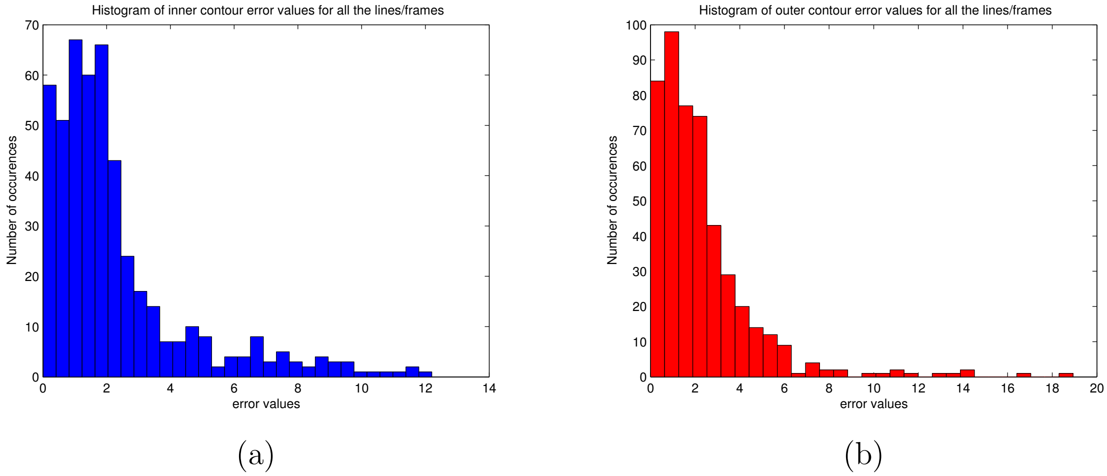
(a) (b)

Figure 4.13: PCD - Root mean squared error (RMSE) of boundary estimation (a) for all the frames (b) for all the lines



(a) (b)

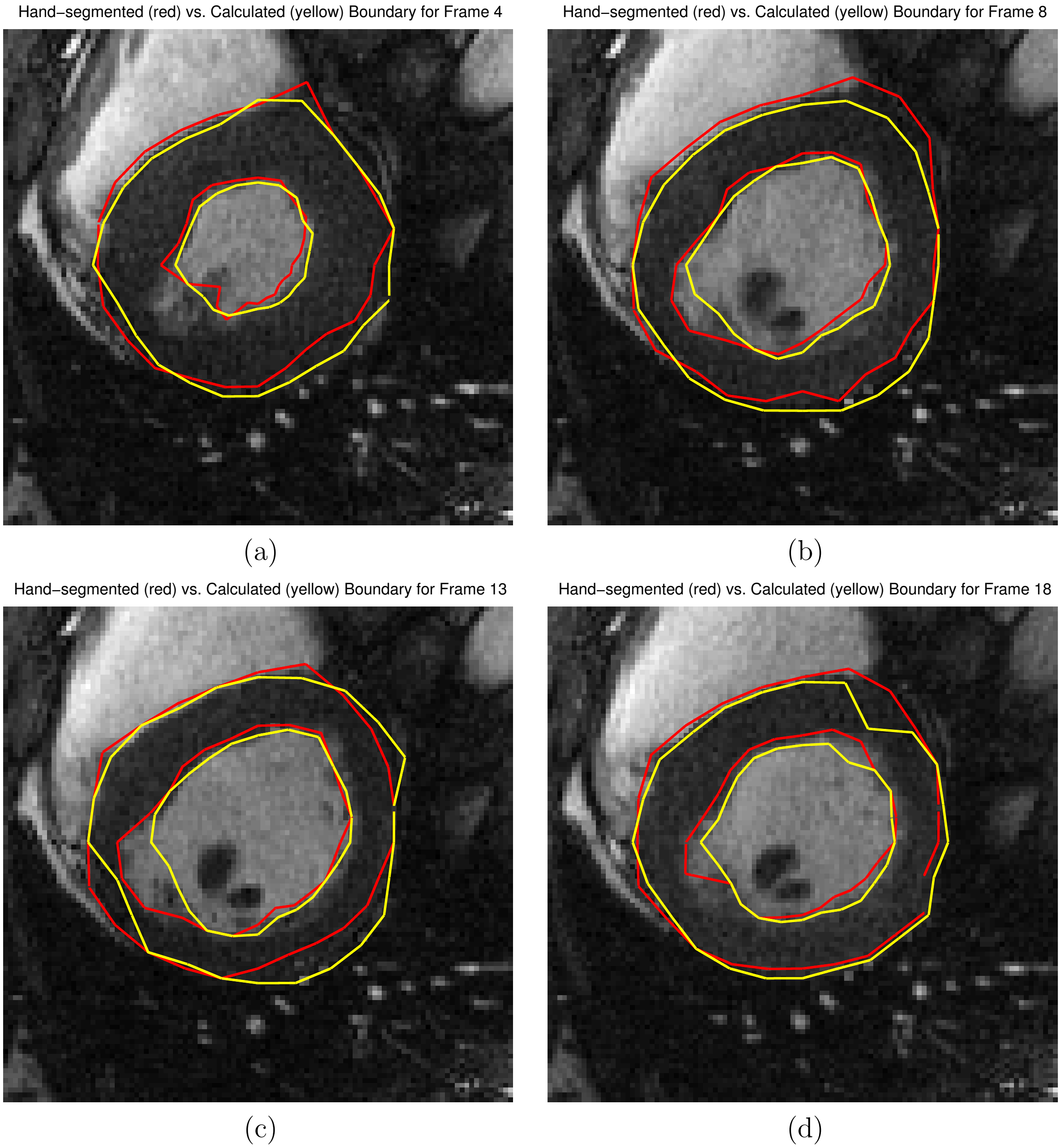
Figure 4.14: PCD - Histograms representing the occurence frequency of each error value (a) inner contour, (b) outer contour



(a)

(b)

Figure 4.15: ICR - Histograms representing the occurence frequency of each error value (a) inner contour, (b) outer contour



Hand−segmented (red) vs. Calculated (yellow) Boundary for Frame 4

(a)

Hand−segmented (red) vs. Calculated (yellow) Boundary for Frame 8

(b)

Hand−segmented (red) vs. Calculated (yellow) Boundary for Frame 13

(c)

Hand−segmented (red) vs. Calculated (yellow) Boundary for Frame 18

(d)

Figure 4.16: Comparing derived contour (shown in yellow) with hand-segmented contour (shown in red) for (a) frame 4, (b) frame 8, (c) frame 13, (d) frame 18.

Chapter 5

Conclusion

Cardiac MR images generally have better resolution and contrast than other imaging modalities such as ultrasound. Additionally, they have the potential to provide valuable clinical information such as cardiac structure and function measurements. However, there are some limitations in widespread use of MRI in clinical applications, the most important of all being the fact that during each cardiac MRI acquisition, more than 100 static 2D scans are generated. In order to evaluate cardiac function measures such as ejection fraction, one has to segment the LV in each of the slices and time frames. Manual segmentation of the LV is very labour-intensive and it is considered to be the bottleneck of the MRI process. Therefore, any application that faciliates semi- or fully automatic segmentation of CMR images will be beneficial to the medical imaging community.

In this work, we developed an algorithm to segment and track the left ventricle in a set of cine MRI, consisting of multiple slices of heart from the apex to the base, each in 20 phases. The focus of our work is to track the LV in a 2D+T data set, consisting of consecutive time frames of a single slice. Some of the challenges of working with MR images include:

- low-contrast in some areas of pericardium adjacent to the lungs and fatty tissue,
- quality degradation by artifacts due to blood flow or respiratory motion, and

- papillary muscles which are located close to the LV and have similar intensity as the myocardium.

Due to these challenges, it is hard to derive estimates based on gradient information alone. We have taken a novel appraoch to this problem, making use of particle filters in a Bayesian framework. A set of radial lines are defined, starting at approximately the centre of the LV. The objective is to estimate intersection of inner and outer boundaries with each one of the radial lines, with particles representing the underlying density probabilities for each line. The algorithm is divided into two main phases, primary contour derivation and iterative contour refinement. The particles are weighted based on a likelihood function, which is defined using the directional derivative values along each line of the frame. The probability distribution estimated from the first phase, is used as a prior to the iterative phase. We also employ the estimates for neighbouring lines from the previous iteration to impose some constraint on the estimation process in the current iteration. We have demonstrated that, given the right set of parameters, the inner and outer contours are detected correctly in most of the frames. We believe that with some minor modifications, this algorithm can be fully automated. In general, our algorithm yields good performance and it is comparable with hand-segmented data.

Contrary to other methods, such as contour-based or shape-based models discussed in Section 2.3, our method does not rely on a complete manual initialization of the first frame. Instead, only the boundary positions of the first line are specified. We also do not make use of training sets or models in our approach. As mentioned earlier, performance of level set methods is degraded by poor image qualities and non-existing boundaries, as the iteration may not converge. We address this problem by coupling the boundary estimations for each line with its neighbours, compensating for low-gradient boundaries. Additionally, level set methods update all the points on the contour at each iteration which makes them slow compared to our method.

Performance of our system depends on a lot of factors, including the number of particles used and more importantly the number of iterations involved in each case. The number of particles depends on the resolution of the image and the length of the defined radial lines. We observed that increasing the number of particles from 60 to 200 did not have a major impact on the overall performance of the system. One of the major challenges of this project was finding a set of parameters that would work in every single case. We have experimented with different set of parameters and presented the effects of varying such parameters in the results chapter. Generally, most of these parameters were found intuitively or experimentally. Learning from our current method, we believe that some of these parameters can be dynamically adjusted based on various measurements.

We also tested different methods or enhancements at various stages of our project. For instance, while sampling enhancement might yield great results on a certain line, it may have the opposite effect on another line of the same frame, or another frame. Therefore, we do not believe that the current sampling enhancement is suited for our application. We also experimented with KL-divergence as the line convergence criteria and did not find it sufficient for our case either. Another lesson learnt from our implementation is that since we are estimating the two boundaries jointly, an erroneous estimate for one contour has a direct effect on estimates for another contour as shown in Figure 4.12. In the future, we can try to assign confidence measures to each one of the estimates individually, and adjust the level of joint contour estimations based on these confidence measures.

5.1 Future Work

We discussed some of the advantages and disadvantages of our project in the previous section. There are numerous ideas that can be implemented based on the current framework. We assume that the data we are processing is causal and we provide boundary estimations using previous frames in forward propagation manner. If this analysis is performed offline, a more comprehensive smoothing in both forward and backward directions could be implemented. Having structured noise, such as papillary muscles, may necessiate use of a model for future stages of the project.

Currently, a stochastic motion model is used in prediction stage of particle filters. Having a model of heart motion would certainly have a positive impact on the performance of this system. We also believe that the current convergence criteria should be improved. An extension of the KL-divergence, or other convergence criteria maybe worth while investigating.

This thesis can be extended to analyze 4D data sets, including the previous and past slices available for the refinement step as well. Software capabilites like parallel processing or multithreading can be utilized to improve the performance of the system significantly.

References

[1] American Heart Association. Heart disease and stroke statistics - 2006 update. online, September 2006. http://circ.ahajournals.org/cgi/content/short/113/6/e85.

[2] Canadian Institutes of Health Research. Heart disease. online, September 2005. http://www.cihr-irsc.gc.ca/e/28901.html.

[3] J. Earls, V. Ho, T. Foo, E. Castillo, and S. Flamm. Cardiac MRI: Recent progress and continued challenges. *Journal of Magnetic Resonance Imaging*, 16:111–127, 2002.

[4] A.F. Frangi, D. Rueckert, and J.S. Duncan. Three-dimensional cardiovascular image analysis. *IEEE Transactions on Medical Imaging*, 21(9):1005–1010, 2002.

[5] J. Rogers. Cardiovascular physiology. Online, 1999. http://www.nda.ox.ac.uk/wfsa/html/u10/u1002_01.htm.

[6] W.J. Germann and C.L. Stanfield. *Principles of Human Physiology.* Benjamin/Cummings, 2004.

[7] Texas Heart Institute. Online, 2006. http://texasheart.org/HIC/Anatomy/anatomy2.cfm.

[8] K.M. Das. Cardiac MR: One-stop shop. *HEART VIEWS*, 5(1):17–23, 2004.

[9] D.C. Noll. A primer on mri and functional mri.

[10] S. Sola, R.D. White, and M. Desai. MRI of the heart: Promises fulfilled? *Cleveland Clinic Journal of Medicine*, 73(7):663–671, July 2006.

[11] J.S. Suri. Computer vision, pattern recognition and image processing in left ventricle segmentation: the last 50 years. *Pattern Analysis and Applications*, 3(3):209–242, 2000.

[12] WE. Higgins and EL. Ritman. 3-d image-enhancement technique for volumetric cardiac images. In AC. Bovik and WE. Higgins, editors, *SPIE, Biomedical Image Processing*, volume 1245, pages 159–170, May 1990.

[13] G.I. Sanchez-Ortiz, J.A. Noble, G.J.T. Wright, J. Feldmar, and M. Mulet-Parada. Automated lv motion analysis from 3d echocardiography. In *Medical Image Understanding and Analysis*, pages 85–88, 1999.

[14] M. Lynch, O. Ghita, and P. F. Whelan. Automatic segmentation of the left ventricle cavity and myocardium in mri data. *Elsevier Computers in Biology and Medicine*, 34(4):389–407, March 2006.

[15] M. Kass, A.P. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, January 1987.

[16] A. Blake and M. Isard. *Active Contours: The Application of Techniques from Graphics,Vision,Control Theory and Statistics to Visual Tracking of Shapes in Motion.* Springer-Verlag, Secaucus, NJ, USA, 1998.

[17] D.R. Cheung. Motion segmentation incorporating active contours for spatial coherence. Master's thesis, University of Toronto, 2004.

[18] T. McInerney and D. Terzopoulos. Deformable models in medical image analysis: a survey. *Medical Image Analysis*, 1(2):91–108, June 1996.

[19] D.N. Davis. The application of active contour models to mr and ct images. Technical report, Medical Vision Group, University of Birmingham, Edgbaston, Birmingham, UK, 1995.

[20] L. Spreeuwers and M. Breeuwer. Detection of left ventricular epi- and endocardian borders using coupled active contours. *International Congress Series*, 1256:1147–1152, 2003.

[21] Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.

[22] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer-Verlag, Secaucus, NJ, USA, 2003.

[23] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International Journal of Computer Vision*, 22(1):61–79, 1997.

[24] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham. Active shape models–their training and application. *Computer Vision & Image Understanding*, 61(1):38–59, January 1995.

[25] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active appearance models. In H. Burkhardt and B. Neumann, editors, *Proceedings of European Conference on Computer Vision*, volume 2, pages 484–498, 1998.

[26] S. Ordas, L. Biosrobert, M. Huguet, and A.F. Frangi. Active shape models with invariant optimal features (iof-asm) application to cardiac mri segmentation. *Computers in Cardiology*, 30:633–636, 2003.

[27] A. Tsai, A. Yezzi, W. Wells, C. Tempany, D. Tucker, A. Fan, E. Grimson, and A. Willsky. A shape-based approach to the segmentation of medical imagery using level sets. *IEEE Transactions on Medical Imaging*, 22(2):137–154, February 2003.

[28] R.M. Lapp, M. Lorenzo-Valdes, and D. Rueckert. 3d/4d cardiac segmentation using active appearance models, nonrigid registration, and the insight toolkit. In C. Barillot, D.R. Haynor, and P Hellier, editors, *Proceedings of the seventh International Conference on Medical Image Computing and Computer-Assisted Intervention*, volume 1, pages 419–426, Saint-Malo, France, September 2004. Springer-Verlag.

[29] X. Huang, Z. Li, and D. Metaxas. Learning coupled prior shape and appearance models for segmentation. In *Proceedings of the seventh International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 60–69, Saint-Malo, France, September 2004. Springer-Verlag.

[30] J.M.B Dias and J.M.N. Leitao. Wall position and thickness estimation from sequences of echocardiographic images. *IEEE Transactions on Medical Imaging*, 15(1):25–38, February 1996.

[31] J. Weng, A. Singh, and M.Y. Chiu. Learning-based ventricle detection from cardiac mr and ct images. *IEEE Transactions on Medical Imaging*, 16(4):378–391, August 1997.

[32] W. Sun, M. etin, R. Chan, V. Reddy, G. Holmvang, V. Chandar, and A. Willsky. Segmenting and tracking the left ventricle by learning the dynamics in cardiac images. In *Information Processing in Medical Imaging*, number 3565 in Lecture Notes in Computer Science, pages 553–565, Berlin Heidelberg, 2005. Springer-Verlag.

[33] J. Sénégas, T. Netsch, C.A. Cocosco, G. Lund, and A. Stork. Segmentation of medical images with a shape and motion model: A bayesian perspective. In *Computer*

Vision and Mathematical Methods in Medical and Biomedical Image Analysis, number 3117 in Lecture Notes in Computer Science, pages 157–168, Berlin Heidelberg, 2004. Springer-Verlag.

[34] D.J. Fleet and A.D. Jepson. Tracking lecture notes, Jan 2004.

[35] The Kalman Filter. online, 1997. http://www.cs.unc.edu/ welch/kalman/.

[36] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. In *IEEE Transactions on Signal Processing*, volume 50, pages 174–188, Piscataway, NJ, USA, Feb 2002. IEEE Signal Processing Society.

[37] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter, Particle Filters for Tracking Applications.* Artech House, Boston, MA, USA, 2004.

[38] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.

[39] C.M. Bishop. *Neural Networks for pattern recognition.* Oxford University Press, 1995.