Semi-Automatic Delineation of the Mitral Valve from Clinical Four-Dimensional Ultrasound Imaging

A dissertation presented
by

Robert Joseph Schneider

to
The School of Engineering and Applied Sciences
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy
in the subject of

Engineering Sciences

Harvard University
Cambridge, Massachusetts
May 2011

Thesis Advisor

Robert D. Howe

Author

Robert Joseph Schneider

Semi-Automatic Delineation of the Mitral Valve from Clinical Four-Dimensional Ultrasound Imaging

Abstract

Successful clinical treatment of mitral valve disease is dependent on understanding the complexities of patient-specific valve behavior. Mechanical models of the mitral valve, designed to predict valve closure and generated using patient-specific measurements, have proven to be effective tools for studying valve behavior. To be clinically feasible, these models need to be generated from ultrasound images, which are commonly acquired during the diagnosis and treatment of patients. However, methods of generating accurate models from ultrasound data using minimal user-input and interaction are not available.

This work presents methods by which a patient-specific mitral valve model (annulus and leaflets) can be generated from three-dimensional ultrasound. The method first identifies the mitral annulus in a user-specified frame just after valve closure using only a roughly placed valve center point. We find the annulus as the location where the thin leaflet tissue intersects the thicker surrounding tissue. Comparing the results to manual delineations made by experts, we found an average RMS error of 1.81±0.78mm. A valve state (open versus closed) predictor and a constrained optical flow algorithm are then used to segment the annulus in the remaining frames in the ultrasound sequence. We compared the automated tracking results to manual delineations made by experts and found an average RMS difference of 1.67±0.63mm. The location of the leaflets in a user-specified frame just before valve closure is then automatically found by using the annulus as a means of limiting segmentation efforts. We compared the segmented leaflets to manual tracings and found an average difference of 0.76±0.65mm. We then use the leaflet geometry and four-dimensional (4D) annulus in an active surface approach to track the leaflets during valve closure, which for the first time allows for the segmentation of an accurate and detailed coaptation region.

This work additionally describes methods to reconstruct from standard clinical acquisitions high temporal resolution ultrasound sequences required to track the progression of valve closure and generate an accurate coaptation region. The reconstruction method includes a real-time image-based rigid registration method for ultrasound volumes which we use to stabilize images in the presence of respiration and other small movements of the probe.

Contents

Acknowledgments

I have been blessed in that for the past several years I have been surrounded by incredibly intelligent and talented individuals who have helped, humbled, and inspired me along the way. This work would not have been possible without their involvement, and for their time and support, I am extremely grateful.

I would first like to thank my advisor, Rob Howe, who provided the perfect blend of guidance and independence. He taught me what I needed to know, and taught me to teach myself that which no one could teach me. I truly appreciate all of his time, effort, patience, and mentorship. I also want to thank advisory committee members Todd Zickler and Hanspeter Pfister for lending me their valuable time and expertise.

My research has afforded me the opportunity to work at Children's Hospital Boston with arguably the finest researchers and clinicians in cardiology and cardiac surgery. Two individuals in particular stand out as playing an especially important role in this work. Doug Perrin has provided invaluable guidance, expertise, counseling, and friendship, and has never shied away from my undoubtedly torturous user studies. Dr. Gerald Marx has also been particularly instrumental in my work, as he not only tirelessly provided me with the necessary clinical data, but also found the time in his busy clinical schedule to provide the clinical insight necessary to motivate my research. To both of these individuals, I am extremely grateful. I would like to also express a special thanks to Dr. Nikolay Vasilyev for his time, patience, and clinical expertise, and for being a willing participant in several user studies, and also to Dr. Pedro del Nido for his time and advice, and for allowing me to work in his lab and invade his operating room to collect clinical data.

This work would not have been as tolerable without the help and welcome distraction provided by past and present Biorobotics Lab members Laura Brattain, Frank Hammond, Leif Jentoft, Sam Kesner, Paul Loschak, and Mahdi Tavakoli. A special thanks to Pete Hammer, Petr Jordan, Neil Tenenholtz, and Shelten Yuen for their friendship, guidance, and counseling, and my most sincere thanks to Pete and Shelten for helping make the transition to fatherhood just a little easier.

It is often said there is a special place in heaven for the spouses of graduate students, and I believe this could not be more true than in the case of my wife Deb. For her steadfast love and support, for everything she has sacrificed to allow me to pursue this goal, and for the gift of our beautiful daughter Abigail, I am forever in her debt.

Dedicated to my wife, Deb.

Chapter 1

Introduction

1.1 Background and Motivation

Visualization and characterization of the mitral valve is especially challenging given its complexity, fast movement, and limitations in current clinical imaging technologies. Oftentimes, acquiring detailed information requires extensive imaging and image processing, which can be time consuming using current methods. This often forces clinicians to make a compromise between speed and completeness of information. However, of the four valves in the heart, the mitral valve is more often associated with disease [95, 68] which requires surgical intervention [54]. It has also been shown that surgical repair of the mitral valve is preferred over replacement [26, 94, 64]. Therefore, detailed information about the mitral valve and surrounding structures is increasingly necessary for a successful clinical outcome.

The mitral valve is a thin leaflet structure residing between the left atrium and left ventricle in the heart (Figure 1.1), and is responsible for maintaining the unidirectional flow of oxygenated blood from the lungs to the rest of the body. It consists of two leaflets, anterior and posterior, which are attached at the base to the mitral valve annulus (Figure 1.2). The opposite ends of the leaflets are attached via chords to a pair of papillary muscles,

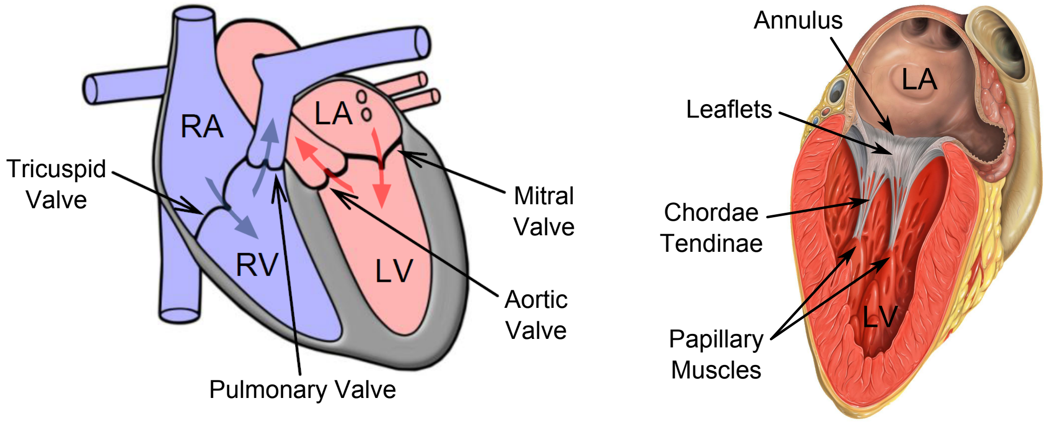


Figure 1.1: (*Left*) Direction of blood flow through the chambers of the heart (Image Source: Peter Hammer, PhD). Deoxygenated blood flows into the right atrium (RA), through the tricuspid valve, into the right ventricle (RV), and then through the pulmonary valve to the lungs. Oxygenated blood then flows back from the lungs into the left atrium (LA), through the mitral valve, into the left ventricle (LV), and then through the aortic valve to the rest of the body. (*Right*) Structures of the mitral valve (Image source: Patrick J. Lynch, medical illustrator; C. Carl Jaffe, MD, cardiologist). The mitral leaflets are anchored to the rest of the heart at the annulus and restricted from flapping into the left atrium by the chordae tendinae (chords) which are anchored to the heart at the papillary muscles.

thereby preventing the leaflets from opening into the left atrium. Common pathologic conditions associated with the mitral valve are prolapse or regurgitation. Mitral prolapse is characterized by a bulging of the leaflets towards the left atrium, a condition that if left untreated can lead to mitral regurgitation. Mitral regurgitation occurs when the leaflets do not fully coapt during valve closure, thereby allowing blood to flow backwards from the left ventricle to the lungs. This results in less blood pumped to the rest of the body and renders the heart less efficient. These conditions are often caused by ischemic heart disease, dilated cardiomyopathy, rheumatic valve disease, or infection [32].

In an effort to better understand mitral valve behavior, many studies have attempted to model the mechanics and dynamics of the mitral valve, surrounding fluid, and/or surrounding structures during valve closure [46, 17, 52, 47, 100, 32, 33, 10]. These models are generally finite-element or mass-spring models designed to simulate the closing of the

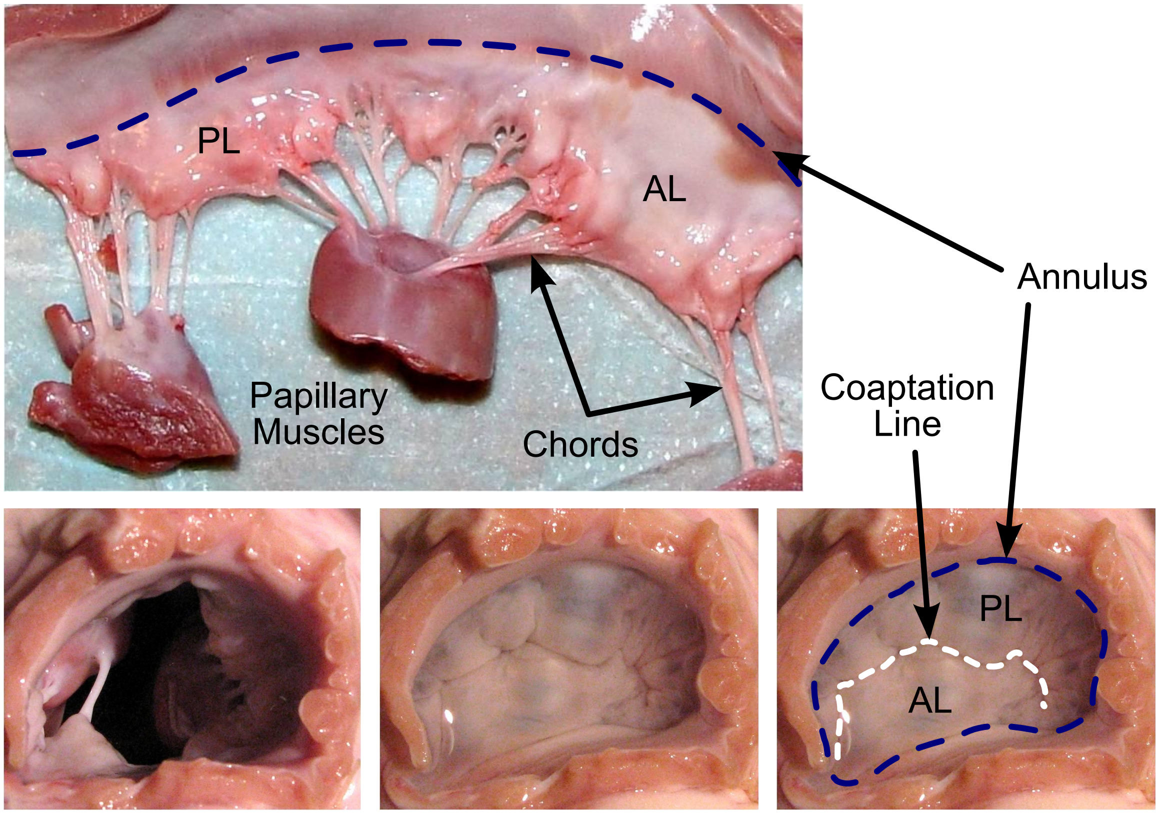


Figure 1.2: Photos of an excised porcine mitral valve. (*Top*) The mitral valve split open to expose the annulus, anterior leaflet (AL), posterior leaflet (PL), chords, and papillary muscles. (*Bottom Left*) Atrial view of an open mitral valve, with the left atrium cut away. (*Bottom Middle*) Atrial view of a closed mitral valve created by artificially pressurizing the left ventricle. (*Bottom Right*) Labeled view of the closed mitral valve showing the location of the annulus, anterior (AL) and posterior (PL) leaflets, and coaptation line. (Images courtesy of Peter Hammer, PhD)

mitral leaflets, and are generally capable of predicting a prolapse or regurgitation. Boundary conditions are defined by the shape and/or location of the leaflets, annulus, chords, and papillary muscles. The development of these models has led to a greater understanding of mitral function and formation of pathologies, and has also presented an opportunity in the area of surgical planning. When patients undergo corrective valve surgery, the heart is put on cardiopulmonary bypass. When on bypass, the heart is drained of blood and is in a deflated, motionless state. It is therefore difficult for a surgeon repairing the valve to predict how it will subsequently function after the heart is taken off bypass. With a model of the mitral valve, however, surgeons could perform a simulated surgery on the valve and be able to better predict mitral function.

To be able to use a predictive model for surgical simulation and planning, the mitral valve model needs to be generated on a patient-specific basis. Because of the time and expense to extract patient-specific anatomy from medical images, current models make assumptions about the shape of the annulus and leaflets, number and location of chords, and the location of the papillary muscles. Previous studies that have attempted to use measurements from actual anatomy have done so using manual segmentations from imaging [100], surgically implanted fiducials at anatomical landmarks [52], or surgically excised hearts from animals [46, 32, 33]. For the purposes of patient specific modeling, however, these methods are insufficient in that they either require extensive user-interaction, are too invasive, or take too much time. Obstacles preventing the use of less invasive and faster methods are poor image quality coupled with the absence of accurate and robust automated methods for mitral valve segmentation.

The most common imaging modality currently used by clinicians to view the heart is 3D ultrasound (3DUS). Images are obtained by placing an ultrasound probe either outside the chest (transthoracic), in the esophagus (transesophageal), or directly on the heart during surgery (epicardial). Using 3DUS, clinicians assess mitral function [79, 38], diagnose pathologies [51, 35, 21, 41, 1, 40, 72, 14, 2, 6], and perform surgical planning [19]. The benefits of ultrasound are that it is a relatively inexpensive, portable, and non-ionizing imaging modality capable of capturing the fast moving valve structures (Figure 1.3). The drawback is that it has a limited field of view and produces noisy images with moderate resolution compared to CT or MRI, making visualization of thin structures such as the mitral valve challenging.

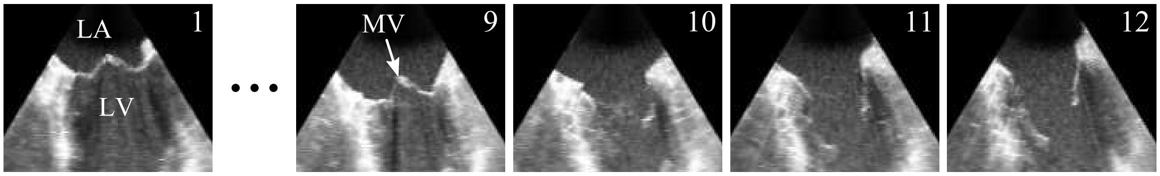


Figure 1.3: Slices from a 3DUS sequence of a prolapsed mitral valve. (LA - left atrium, LV - left ventricle, MV - mitral valve)

1.2 Mitral Annulus and Leaflets from 3DUS

Volume rendering and noise in 3DUS make extracting accurate three-dimensional geometry of the mitral valve difficult. A number of methods have been developed for the leaflets and/or annulus using 2D ultrasound in which manual initializations were tracked throughout a sequence [60, 87, 104, 18, 91, 58, 66]. There are also a couple methods designed to segment and/or track only a portion of the valve (i.e. just the annulus) in 3DUS [107, 98]. However, aside from those that have done manual segmentation the mitral valve from 3D and 4D ultrasound sequences [104], there are only a few groups that have come close to generating patient-specific mitral valve models. The method by Shang *et al.*, 2008 was an intensity based level set segmentation method that was able to segment the mitral leaflets from a 3DUS volume [85]. Unfortunately, the resulting segmentation included not only the leaflets but everything connected to the leaflets with similar intensity (i.e. the surrounding heart wall) and therefore was unable to isolate the valve from the surroundings. It would also not be able to distinguish between leaflets if the valve was closed. The method by Burlina *et al.*, 2010 used a thin tissue detector to segment the mitral leaflets and a level set segmentation method to segment the left ventricle [10]. These segmentations were then combined to generate a patient-specific mitral valve geometry. The problems with the presented method, however, are that it required extensive user-interaction and the mitral annulus was "formed" using the assumption of a planar annulus, despite the fact that the annulus has been shown to be saddle-shaped in healthy valves [38]. Lastly, the method by

Ionasec *et al.*, 2010 is a sophisticated system that fit a mitral and aortic valve model to 3DUS data using machine learning techniques and a large database of manually delineated points [37]. The accessibility of this method is limited, however, due to the fact that it took the researchers nearly two years to compile and manually delineate the data necessary for the described methods. The method also appeared to generate a mitral valve model that lacked patient-specific detail, especially when compared to the segmentations in [10], and did not appear to be able to generate a detailed or accurate coaptation region or length.

1.3 Mitral Chords from 3DUS

While the mitral valve annulus and leaflets have been shown to be segmented from ultrasound, there is still no evidence that accurate chordal structure can be found using clinical 3DUS images. There are several reasons why this continues to be a challenge. The first is that chords are extremely thin structures, often less than 1mm in diameter. This is on the order of the resolution of 3DUS volumes, which are typically 0.5–0.7mm/voxel. Also, the chords are actually a complicated and dense network of tissue, making segmentation of specific cords a challenge again due to the resolution of 3DUS. The location and orientation of the chords also makes it challenging to image them using ultrasound, as they typically appear in shadowed regions formed from either the left ventricle wall, the mitral leaflets, or other chords. To elucidate the challenges with chordal imaging in 3DUS, Figure 1.4 shows the chordal structure of the same valve imaged using both micro CT and 3DUS. For these reasons, segmentation of the chords using clinical 3DUS remains a challenge and is not included in this work, but will be addressed in future work.

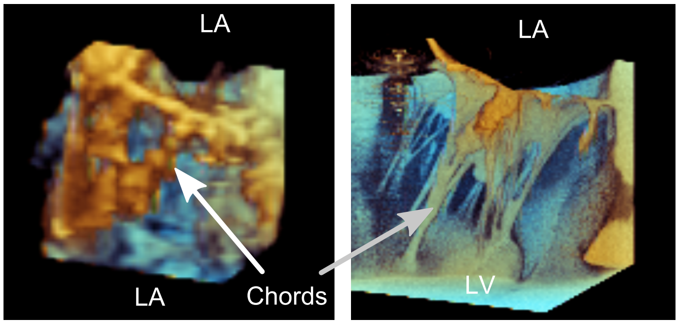


Figure 1.4: The mitral valve chords from the same valve as seen in (*Left*) 3DUS (trans-esophageal, 4-chamber approach) and as seen in (*Right*) micro CT. The lack of definition of the chords in 3DUS explains why accurate chordal segmentation from clinical 3DUS remains a challenge. (LA - left atrium, LV - left ventricle)

1.4 Thesis Contributions

The contributions of this work are an image processing and valve segmentation pipeline that provide for the first time a means by which detailed heart valve information can be ascertained from standard clinical echocardiography acquisitions.

The image processing pipeline is a series of methods designed to reconstruct a sequence with millisecond frame resolution from a standard clinical three-dimensional echocardiography acquisition. Contrary to previous work in this area, this is accomplished with minimal deviation from standard clinical imaging protocol and does not require any additional hardware or software. Specific challenges that are addressed in this work are how to correct for temporal distortion within each frame, and also how to compensate for small movements of the heart relative to the probe during long acquisitions (i.e. respiration compensation). The reconstruction process is completely automated and made to operate in under a half hour, even for acquisitions of several hundred frames, making the pipeline a valuable tool for both clinical and research purposes.

The valve segmentation pipeline is a sequence of methods which progressively define the components of a valve so that an accurate four-dimensional segmentation is generated from ultrasound imaging. The pipeline is designed to first find the location of the annulus throughout a sequence and to then use this location to help find the leaflets. The annulus is found by first segmenting its location when the valve is closed and then tracking the annulus when the valve is open. The leaflets are then found by first segmenting their location when the valve is open and then tracking their location to all other frames. In this work, the individual methods were catered specifically to the mitral valve, however, the general framework should hold for the other valves in heart. Specific challenges that are addressed in this work are how to semi-automatically segment the annulus for a closed valve, automatically segment the leaflets for an open valve, and automatically track the annulus and leaflets to the remaining frames. The methods were designed to be robust even in the presence of ultrasound noise and anatomic variation, and so it stands to reason that the segmentation pipeline could also be used with CT or MR images, which typically have better spatial resolution and a higher signal-to-noise ratio, provided that the images had sufficient temporal resolution.

To our knowledge, this works shows for the first time how to generate the highest temporal resolution imaging of the heart using standard ultrasound acquisitions, and in using this data for valve segmentation, shows for the first time a high temporal resolution segmentation of the mitral valve anatomy. This has in turn allowed for the first time an accurate visualization and segmentation of the progression of valve closure, and consequently results in an accurate delineation of the coaptation region and length. In being able to generating such a detailed, four-dimensional, patient-specific valve segmentation, this is also the first time that a mitral valve annulus and leaflets suitable for use in mechanical models has been generated from ultrasound, and also provides a means by which the models can be validated using ultrasound data.

1.5 Thesis Overview

This thesis is presented as a collection of methods necessary for generating a detailed, patient-specific, high temporal resolution, four dimensional mitral annulus and leaflet segmentation.

Chapter 2 describes a method by which the mitral annulus is found in a single 3DUS frame showing a closed mitral valve using only a single user-defined point near the center of the valve. The method operates by assuming that the thin mitral leaflet tissue can first be found using a graph-cut algorithm, and that the annulus is then the location where the thin tissue intersects the thicker surrounding tissue. The method is validated by comparing the resulting annulus to both manual segmentations made by a group of experts on clinical images and to delineations made by an expert with an electromagnetically tracked device on a fully exposed mitral annulus in a water tank.

Chapter 3 then describes how to use the method described in Chapter 2, along with a valve state predictor and a constrained optical flow method, to track the annulus throughout an ultrasound sequence. The method is again validated by comparing the results to manual delineations made by a group of experts using clinical images.

Chapter 4 outlines a method by which the mitral valve leaflets are automatically segmented in a frame showing an open mitral valve. The method requires as an input the location of the annulus in the designated frame, which can be generated using the method described in Chapter 3. Knowing that the mitral leaflets are connected to the annulus and rotate about its location, the location and extent of the mitral leaflets are estimated and refined using both graph-cut and level-set segmentation algorithms. The method is validated by comparing the segmented leaflets to manual tracings made by an expert.

Chapter 5 describes how to improve the temporal resolution of an ultrasound sequence. This is necessary because in order to track the leaflets and accurately determine a coaptation location and extent, the progression of valve closure needs to be observed, which is not possible using native ultrasound acquisitions. The method operates by making varying simplifying assumptions about ultrasound volume formation, and by also exploiting the periodicity of cardiac motion.

Chapter 6 describes a real-time, image-based rigid registration algorithm for 3DUS volumes that is essential for the reconstruction pipeline presented in Chapter 5. The method is a feature-based registration algorithm that is similar to existing methods, but by taking into account the nature and behavior of real-time 3DUS data, is able to drastically simplify existing methods so that real-time registration can be achieved. The performance of the method is studied and validated in several studies.

Chapter 7 completes the discussion of mitral valve segmentation from 3DUS by outlining a method by which the mitral valve annulus and leaflets can be segmented and tracked during mitral valve closure. The method requires the high temporal resolution imaging generated using the methods described in Chapter 5, and requires as input the annulus and leaflet segmentations generated using the methods in Chapters 3 and 4. The leaflet segmentation is then used in an active surface formulation to track the leaflets from frame to frame.

Chapter 8 discusses the broader impacts of this research in the areas of ultrasound image processing and valve segmentation. In particular, the impact of the designed ultrasound reconstruction and registration methods on areas of research such as super-resolution and denoising are discussed. Also, the usefulness of the valve segmentation framework in the context of understanding other valves other than the mitral is discussed, as well as the possibilities of expanding the framework to include real-time analysis. Lastly, future work both in mitral valve segmentation and in the general area of image processing of 3D echocardiography is outlined.

Chapter 2

Three-Dimensional Mitral Annulus Segmentation for Closed Valves

Despite widespread use of the annulus in both clinical and research applications, available methods to extract an accurate and reproducible geometry remain limited. One of the most common methods used in research applications is to segment and track implanted fiducials, such as tracking radiopaque markers with fluoroscopy [25, 13, 93, 6] or tracking sonomicrometry transducers [27]. This is an invasive approach not feasible in a clinical setting. Another common method is manual segmentation of images [21, 41, 40, 72, 104, 14, 74], which in addition to being tedious and time consuming, is also prone to inaccuracies. One reason is that the usual practice is to pick points in 2D slices taken from a 3D volume, so the user only has access to a portion of the available information at any given step, forcing the user to mentally interpolate 3D information. The methods presented in [76] and [49] attempt to correct for the deficiencies in these manual slice-based segmentations, but do so by smoothing the segmentations without referring back to the original 3D data from which they were made.

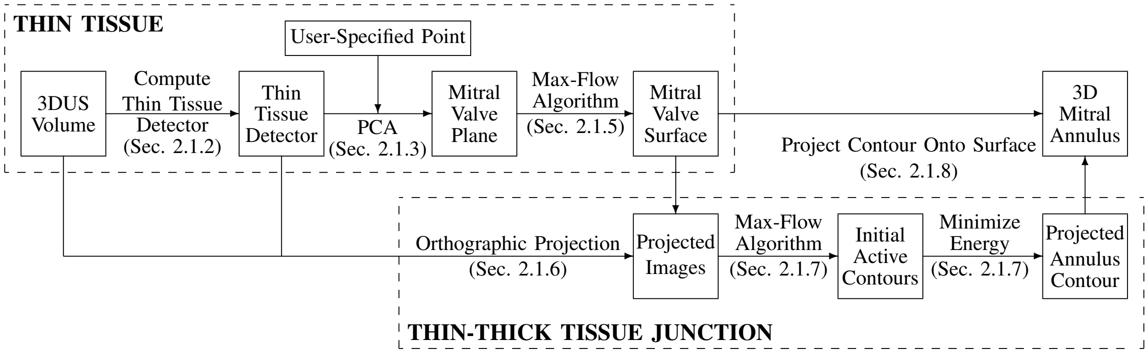

THIN TISSUE
User-Specified Point
3DUS Volume
Compute Thin Tissue Detector (Sec. 2.1.2)
Thin Tissue Detector
PCA (Sec. 2.1.3)
Mitral Valve Plane
Max-Flow Algorithm (Sec. 2.1.5)
Mitral Valve Surface
3D Mitral Annulus
Project Contour Onto Surface (Sec. 2.1.8)
Orthographic Projection (Sec. 2.1.6)
Projected Images
Max-Flow Algorithm (Sec. 2.1.7)
Initial Active Contours
Minimize Energy (Sec. 2.1.7)
Projected Annulus Contour
THIN-THICK TISSUE JUNCTION

Figure 2.1: Flow chart for the mitral annulus segmentation algorithm designed for closed valves which locates the annulus by first finding the thin leaflet tissue and then finding where the thin tissue meets the thick tissue of the surrounding heart wall. Specific details for the respective processes can be found in the indicated sections.

Semi-automatic methods to delineate the 3D annulus from 3DUS are presented in [107] and [99]. The former method segments the annulus by compiling semi-automated annulus point delineations in 2D slices into a 3D annulus structure. As in the case of manual segmentation, this method does not take into account information in neighboring slices, so it suffers from inaccuracies and spatial inconsistencies. The latter method segments the annulus as a consequence of fitting an entire mitral valve model to 3DUS data using learning techniques, which consequently requires a training database of manually delineated points. The resulting accuracy of the method for the purpose of patient-specific annulus segmentation is unclear, as comparisons are only made to published population valve dimensions.

We present in this chapter an accurate and robust segmentation algorithm (Fig. 2.1) designed to segment the mitral annulus in an ultrasound volume containing a closed valve. The algorithm begins by first finding a surface at the location of the thin leaflets. The surface is constructed near a user-provided point using a thin tissue detector and the max-flow algorithm. The algorithm then locates the annulus by finding where at the surface the thin leaflet tissue meets the thicker tissue of the surrounding heart wall. This is done using the max-flow algorithm and active contour methods in projection images created from the surface and surroundings. A 3D mitral annulus contour is computed by projecting the annulus contour in the images back onto the mitral leaflet surface.

The details of the algorithm are described in Section 2.1. An analysis of the algorithm is then presented in Section 2.2, which includes studies which validate the algorithm's accuracy (Sections 2.2.1 and 2.2.2), and an analysis of the sensitivity of the algorithm to the user-provided point (Section 2.2.3).

2.1 Algorithm Design

2.1.1 Ultrasound Data

The acquired data for this study was in the form of full volume reconstructions, allowing for the visualization of the entire mitral valve in a single 3DUS volume (iE33 Echocardiography System with transesophageal, X7-2t, and transthoracic, X7-2, probes, Philips Healthcare, Andover, MA, USA). The condition of the valves ranged from normal to varying types and degrees of pathology. While the position and orientation of a valve within a volume varied, the segmentation algorithm requires that the thin tissue of the leaflets can be seen across the entire valve.

From each 4D data set (3D + time), a single ultrasound frame containing a closed valve was selected. The frame was chosen at or near peak systole, as this is a time in the cardiac cycle when the valve is closed and moving slowly. Restricting the selection to closed valves enabled accurate computation of a surface at the location of the leaflets. The dimensions of the volume were roughly 200×200×200 voxels, with a resolution on the order of 0.5-0.75 mm per voxel. In this chapter, the selected 3DUS volume is referred to as Ω, and individual voxel locations as $\mathbf{v}_i$, where each $\mathbf{v}_i$ has corresponding (x_i, y_i, z_i) coordinates. For Ω with N voxels, $i \in \{1, ..., N\}$.

2.1.2 Thin Tissue Detector

Delineating the location of the annulus directly from 3DUS is difficult, so the algorithm first segments the location of the leaflets using a thin tissue detector. The thin tissue detector (TTD) highlights thin structures at a particular scale in the 3DUS volume. The TTD is computed using characteristics of the gradient field of both Ω and Ω_F, where Ω_F is a filtered version of Ω, reducing the effects of speckle in the volume. Ω_F is computed by convolving Ω with a Gaussian kernel, $G(\sigma)$, with standard deviation, σ, approximately equal to half the maximum expected thickness of the mitral leaflets, taking into account any abnormal thickness due to disease state. The kernel dimension is $4\sigma+1$ to include two standard deviations.

The TTD is computed using the results of three computations. The first of these

$$\Theta\left(\mathbf{v}_i\right)=\frac{\sum_{a=1}^{J-1}\sum_{b=a+1}^{J}\arccos\left(\frac{\nabla\Omega_F(\mathbf{v}_a)\cdot\nabla\Omega_F(\mathbf{v}_b)}{|\nabla\Omega_F(\mathbf{v}_a)||\nabla\Omega_F(\mathbf{v}_b)|}\right)}{\sum_{j=1}^{J-1}j} \tag{2.1}$$

is a measure of the average angle between gradient vectors of Ω_F within a neighborhood, where ∇ denotes the gradient operator. A neighborhood at $\mathbf{v}_i$ is defined as the set of all neighboring voxels, $\mathbf{v}_j$, which are contained within the cube of side length p centered at voxel $\mathbf{v}_i$, making $j\in\{1,...,J\}$, with $J=p^3$ for a cubical neighborhood. At the location of the leaflets, there will be opposing gradient fields as a result of the dark blood pools above and below the thin, bright leaflets. Therefore, Θ will have a higher value at the leaflets in comparison to more homogeneous regions, such as in the middle of the blood pool or thick tissue regions.

The gradient vectors at the leaflets should be pointing inward, so

$$\Phi\left(\mathbf{v}_i\right)=\sum_{q=1}^{6}\sum_{k=1}^{K}\left(\frac{\nabla\Omega_F\left(\mathbf{f}_k\right)}{\left|\nabla\Omega_F\left(\mathbf{f}_k\right)\right|}\cdot\mathbf{f}_{q,norm}\right) \tag{2.2}$$

measures the flux of the vectors across the neighborhood boundary faces. In computing Φ, the *inward* directed unit normal of each boundary face is denoted $\mathbf{f}_{q,norm}$, where $q\in\{1,...,6\}$

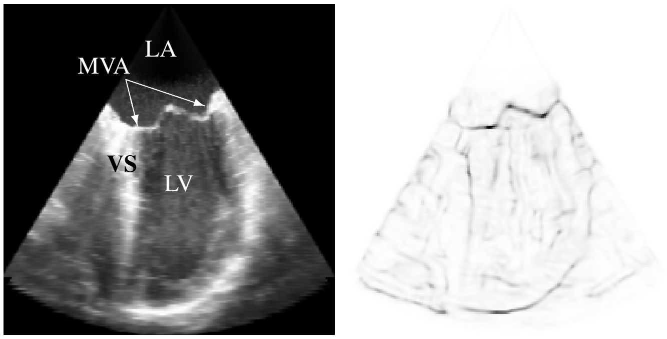


Figure 2.2: *(Left)* Slice from a 3DUS volume of a prolapsed mitral valve showing the location of the mitral valve annulus (MVA), left atrium (LA), left ventricle (LV), and ventricular septum (VS). *(Right)* Corresponding slice from the thin tissue detector (inverted for clarity) computed for the 3DUS volume.

for a cubical neighborhood. Neighbors with a face adjacent to face q are denoted $\mathbf{f}_k$, where $k \in \{1, ..., K\}$ and $K = p^2$ for a cubical neighborhood.

Θ and Φ are computed using directions of gradient vectors but not their magnitude, meaning they contain no information about edges. However, the mitral leaflets are near two strong edges as a consequence of the thin tissue residing between two blood pools. Therefore, to quantify the number and proximity of strong edges to a voxel

$$\Pi = G\left(\sigma\right) * \left|\nabla\Omega\right|. \tag{2.3}$$

The complete TTD is then $\left(\Theta\ \Phi_{[0-1]}\ \Pi\right)_{[0-1]}$, where $(\cdot)_{[0-1]}$ indicates that the values for that term are normalized to the range of 0 to 1. High TTD values indicate that the voxel is likely part of a thin structure and potentially at the location of the mitral leaflets. The TTD computed for a 3DUS image of a prolapsed mitral valve is shown in Fig. 2.2, where the TTD is inverted to better show the highlighted thin tissue regions.

2.1.3 Valve Position & Orientation

To accurately construct a surface at the location of the mitral leaflets, an initial estimation of the position and orientation of the mitral valve relative to the volume is needed. In the first iteration of the algorithm, the position is determined by the user, who is asked to provide a point, $\mathbf{x}_c$, somewhere near the center of the valve but not necessarily on the valve. In subsequent iterations, assuming they are needed, $\mathbf{x}_c$ is defined as the center of the computed annulus from the previous iteration.

Given $\mathbf{x}_c$ near the center of the valve, the orientation of the mitral valve is then estimated by computing a corresponding best-fit mitral valve plane. The idea is to threshold to separate leaflet tissue and blood, then fit a plane to the tissue. This is done by first clustering the values of the TTD from voxels residing in a spherical region of interest of radius r_{pca} centered at $\mathbf{x}_c$ into two clusters using a k-means algorithm. The sample mean and standard deviation of the TTD at voxels for the cluster likely containing the leaflets, which is the cluster with the higher TTD average, are denoted μ_{high} and σ_{high}, respectively. A principle-component analysis (PCA) on the set of voxels $\mathbf{v}_{pca} : \{\mathbf{v}_i | \mathrm{TTD}\,(\mathbf{v}_i) > (\ \mu_{high} - \sigma_{high}\ ) \cap |\mathbf{v}_i - \mathbf{x}_c| \leq r_{pca}\}$ then defines the orientation of the cluster. The mitral valve plane, MV_{plane}, is then the plane passing through $\mathbf{x}_c$ with normal direction $\mathbf{n}_c$ equal to the direction of least variance as determined in the PCA.

2.1.4 Graph Construction and the Max-Flow Algorithm

The annulus segmentation algorithm makes extensive use of the max-flow algorithm [7]. Generically, a graph, $\mathit{\Gamma}{=}\langle V, E\rangle$, is a set of nodes, V, connected by edges, E, which have either a directed or undirected capacity. In the case of the max-flow algorithm, there are also two special nodes called the source and the sink. The max-flow algorithm finds the maximum flow which can originate from the source, flow through the edges of predefined capacity, and enter the sink. In doing so, according to the min-cut/max-flow theorem [22], the max-flow algorithm generates a set of saturated edges called the min-cut which separates the graph into two regions – one containing the source and another

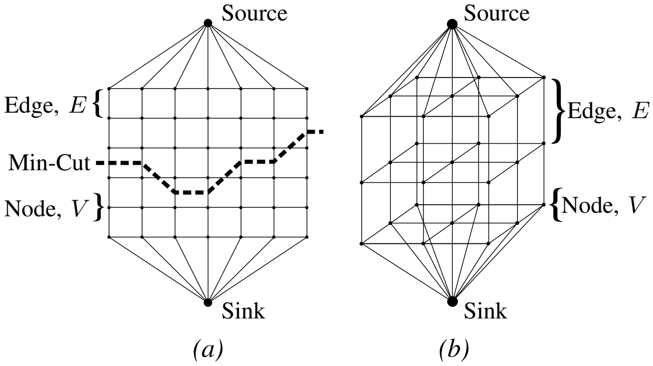


Figure 2.3: Generalized graph structures used in the mitral annulus segmentation algorithm: *(a)* 2D graph with a min-cut example, *(b)* 3D rectilinear graph

containing the sink.

A generalization of the graph construction used in the algorithm is shown in Fig. 2.3, where the 2D and 3D graphs are used to find contours and surfaces, respectively. The graph construction, in particular attaching the source and sink to opposite ends of the graph, allows us to enforce prior knowledge that the min-cut should reside between the opposite ends.

2.1.5 Mitral Leaflet Surface via Max-Flow

With an estimation of the valve position and orientation, a graph can be constructed on which we implement the max-flow algorithm to find a surface at the location of the mitral leaflets. The graph, $\mathit{\Gamma}_{surf}{=}\langle V_s, E_s\rangle$, resides within a cylinder of radius r_{graph} and height $2r_{graph}$ which is centered at $\mathbf{x}_c$ with an axis directed along $\mathbf{n}_c$. The graph consists of nodes V_s located on a rectilinear grid directed along $\mathbf{n}_c$ with a one voxel spacing, and undirected edges E_s which connect the nodes, making $\mathit{\Gamma}_{surf}$ 6-connected except on the edges of the graph. The source connects to all nodes on one face of the cylinder, while the sink connects to all nodes on the opposite face (Fig. 2.4).

To find a surface at the location of the leaflets, we define the edge capacities, $E_{s,p}$

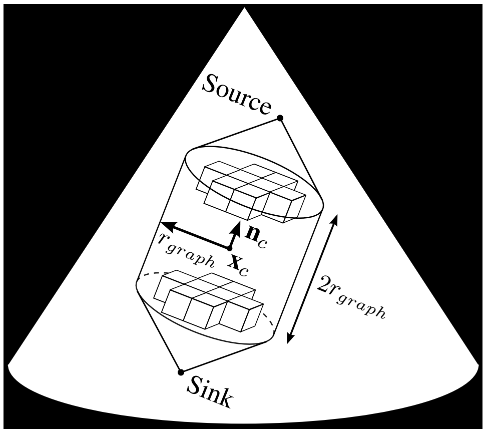


Figure 2.4: Position, orientation, and cylindrical boundary of the 3D rectilinear graph used to find the mitral leaflet surface

and $E_{s,o}$, in (2.4) and (2.5), respectively, such that between connected nodes $V_{s,i}$ and $V_{s,j}$

$$E_{s,p} = \frac{\omega_p}{1 + \alpha_s \left(\mathrm{TTD}\left(V_{s,i}\right) + \mathrm{TTD}\left(V_{s,j}\right)\right)^2} \tag{2.4}$$

$$E_{s,o} = \frac{\omega_o}{1 + \alpha_s \left(\mathrm{TTD}\left(V_{s,i}\right) + \mathrm{TTD}\left(V_{s,j}\right)\right)^2} \tag{2.5}$$

where ω_p, ω_o, and α_s are scalar weights. This form lowers the edge capacity around the leaflets, encouraging the min-cut to be located at the leaflets. $E_{s,p}$ defines the capacity for edges parallel to $\mathbf{n}_c$, and $E_{s,o}$ the capacity for all other edges. This is an anisotropic edge capacity assignment [78] used to control the curvature of the surface: $\omega_p < \omega_o$ encourages a flat surface, while $\omega_p > \omega_o$ allows for a high curvature surface. For those edges where one of the nodes is either the source or the sink, the edge capacity is set to infinity.

The min-cut is found using the max-flow implementation by Kolmogorov [45]. The resulting min-cut defines the mitral valve leaflet surface, MV_{surf}. Assuming $\mathbf{x}_c$ is located near the center of the valve and r_{graph} is sufficiently large, MV_{surf} extends beyond the leaflets and contains the mitral annulus.

2.1.6 Projection Image Formation

The annulus is where the thin leaflets connect to the thicker cardiac walls. We therefore search at and around MV_{surf}, which contains the annulus, for this location. Because the curvature of MV_{surf} was controlled using ω_p and ω_o in the max-flow implementation, MV_{surf} is a regular surface in $\Re^3$ that can be described as the graph of a function, Z_{surf}, described in the $\mathbf{x}'\mathbf{y}'$-plane

$$MV_{surf}\left(x', y'\right) = \left(x', y', \mathrm{Z}_{surf}\left(x', y'\right)\right). \tag{2.6}$$

The $\mathbf{x}'\mathbf{y}'\mathbf{z}'$-coordinate system is centered at $\mathbf{x}_c$ with the $\mathbf{z}'$ axis parallel to $\mathbf{n}_c$ (Fig. 2.5).

The annulus, a closed curve on MV_{surf}, therefore projects to a closed planar curve in the $\mathbf{x}'\mathbf{y}'$-plane. To locate this planar curve, two images, P_{int} and P_{ttd}, are used, where

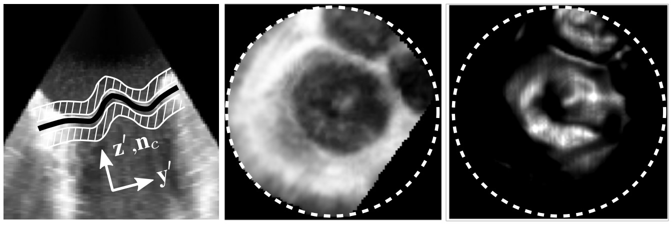
$$\mathrm{P}_{int}\left(x', y'\right) = \sum_{\zeta=\sigma}^{2\sigma} \Omega\left(x', y', \mathrm{Z}_{surf}\left(x', y'\right) \pm \zeta\right) \tag{2.7}$$

$$\mathrm{P}_{ttd}\left(x', y'\right) = \mathrm{TTD}\left(x', y', \mathrm{Z}_{surf}\left(x', y'\right)\right). \tag{2.8}$$

Examples of the images are shown in Fig. 2.5. P_{int} is the value of Ω in regions above and below MV_{surf}, excluding a band of thickness 2σ where the leaflets are located, and provides information about where the tissue surrounding the leaflets resides. P_{ttd} is the value of the TTD at MV_{surf}, and provides information about where the thin tissue resides at the surface. It is clear from the example images of P_{int} and P_{ttd} in Fig. 2.5 that there exists a centrally located dark region in P_{int} and a bright region in P_{ttd}. This region corresponds to the location of the mitral valve in the projection space, making the annulus the border of this region.

2.1.7 Projected Mitral Annulus Contour

The border that we wish to delineate in the projection space, which corresponds to the projected location of the mitral annulus, is a single closed contour which can be



(a) *(b)* *(c)*

Figure 2.5: *(a)* Slice normal to the valve plane from 3DUS of a prolapsed mitral valve (same data as shown in Fig. 2.2) showing MV_{surf} (black) and adjacent regions (white striped) used to form P_{int}, *(b)* intensity projection P_{int}, and *(c)* thin tissue detector projection P_{ttd}. Projection images are only defined within the dotted circles.

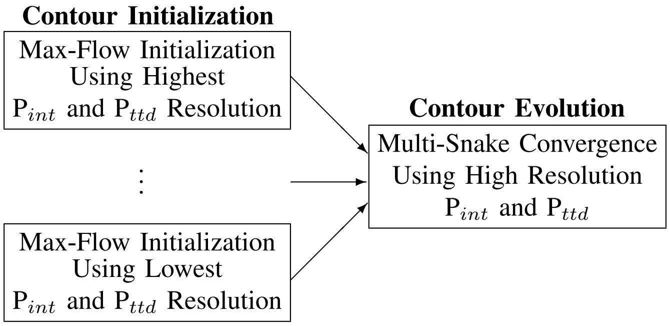

Contour Initialization
Max-Flow Initialization Using Highest P_{int} and P_{ttd} Resolution
⋮
Max-Flow Initialization Using Lowest P_{int} and P_{ttd} Resolution
Contour Evolution
Multi-Snake Convergence Using High Resolution P_{int} and P_{ttd}

Figure 2.6: Contour initialization and evolution scheme for finding the annulus in the projection plane

delineated with such methods as snakes [43] or level sets [84]. These methods, however, are sensitive to initial contour position and generally require a manually initialized contour. To avoid the variability inherent in user input, we developed an automated method which initializes contours by computing min-cuts on 2D graphs using the max-flow algorithm, with edge capacities derived from both P_{int} and P_{ttd} for added robustness. This method is preferred over our preliminary method which initialized generic contours in the projection images [82]. To overcome noise and anatomic variability in P_{int} and P_{ttd}, multiple spatial scales of the projection images are used to construct multiple resolution-specific contours. Treating these contours as snakes and forcing them to a common location in high resolution versions of P_{int} and P_{ttd} produces a single contour at the desired location. A summary of the approach is shown in Fig. 2.6. The advantage of using the method of snakes versus level

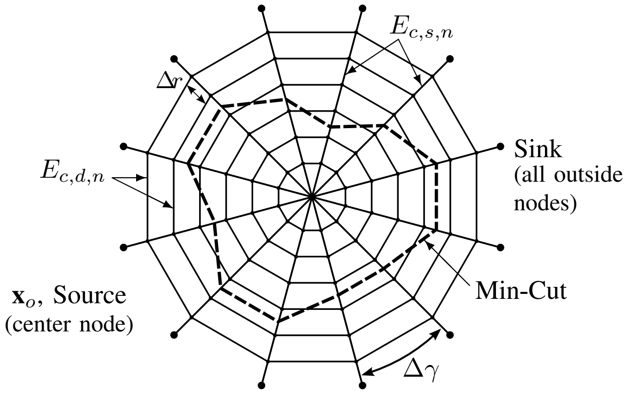


Figure 2.7: Graph used in the max-flow algorithm for contour initialization in the projection space, along with an example of a min-cut

sets is it allows for a simple integration of an attractive energy between contours and does not allow a contour to split during its evolution.

Contour Initialization via Max-Flow

The graph, $\mathit{\Gamma}_{cont}$=$\langle V_c, E_c\rangle$, used in the max-flow implementation to initialize a contour in the projection space is shown in Fig. 2.7. The nodes, V_c, are positioned along rays emanating from a center point, $\mathbf{x}_o$. The rays span the defined projection space and are evenly spaced at an angular offset of $\Delta\gamma$, while nodes are evenly spaced along the rays at an offset of Δr. The nodes nearest to the center connect to the source, while nodes farthest from the center connect to the sink.

To initialize a contour at a specific resolution of the projection images, P_{int} and P_{ttd} are first filtered using a Gaussian, $G(\sigma_n)$, to generate $\mathrm{F}_{int,n}$ and $\mathrm{F}_{ttd,n}$, respectively. For the case where m different contours will be initialized at m different resolutions, $n \in \{1, ..., m\}$. The values of $\mathrm{F}_{int,n}$ and $\mathrm{F}_{ttd,n}$ at the node locations are found using bilinear interpolation and are normalized across all nodes to the range [0-1].

It is assumed that the border of the centrally located regions surrounds the center of the rays, $\mathbf{x}_o$. To find this border, we first compute at the node locations

$$\mathrm{F}_{drv,n} = \left(\lambda_{int}\frac{\partial \mathrm{F}_{int,n}}{\partial R} - \lambda_{ttd}\frac{\partial \mathrm{F}_{ttd,n}}{\partial R}\right)\left(1 - \mathrm{F}_{int,n}\right)^2 \tag{2.9}$$

where λ_{int} and λ_{ttd} are scalar weights and R is the direction along a ray. This is a drive image that will help to define the graph edge capacities, with high values at locations that are most likely the desired border. Partial derivatives of $\mathrm{F}_{int,n}$ and $\mathrm{F}_{ttd,n}$ are computed along the rays because we would like to find the location where $\mathrm{F}_{int,n}$ changes from dark to light as we travel out from $\mathbf{x}_o$ along a ray, rather than light to dark. Similarly, we want to find the location where $\mathrm{F}_{ttd,n}$ changes from light to dark. $\mathrm{F}_{drv,n}$ is scaled by the inverse of $\mathrm{F}_{int,n}$ to encourage the contour to be found near the darker regions of $\mathrm{F}_{int,n}$.

The edge capacities, $E_{c,s,n}$ and $E_{c,d,n}$, of the graph are undirected and defined using $\mathrm{F}_{drv,n}$ such that between connected nodes $V_{c,i}$ and $V_{c,j}$

$$E_{c,s,n} = \frac{\omega_s}{1 + \alpha_c\left(\mathrm{F}_{drv,n}\left(V_{c,i}\right) + \mathrm{F}_{drv,n}\left(V_{c,j}\right)\right)^2} \tag{2.10}$$

$$E_{c,d,n} = \frac{\omega_d}{1 + \alpha_c\left(\mathrm{F}_{drv,n}\left(V_{c,i}\right) + \mathrm{F}_{drv,n}\left(V_{c,j}\right)\right)^2} \tag{2.11}$$

where ω_s, ω_d, and α_c are scalar weights. $E_{c,s,n}$ defines the capacity for edges between nodes on the same ray, and $E_{c,d,n}$ defines the capacity for edges between nodes on different rays. This is done to control the deviation of a contour from a circle: $\omega_s < \omega_d$ encourages a more circular min-cut, while $\omega_s > \omega_d$ allows for a less circular min-cut.

Due to the construction of the graph, nodal resolution varies with distance from the center. Minimizing these variations along the min-cut requires iterating the contour initialization to force the ray center to the centroid of the area contained within the min-cut. For the first iteration, $\mathbf{x}_o$ is the projection of the point $\mathbf{x}_c$ onto the projection space. The final location of the ray center is $\mathbf{x}_{f,n}$, and the final min-cut is contour C_n, where n is specific to the resolution of the projection images.

Contour Convergence via Active Contours

The multiple contours initialized using m different resolutions of P_{int} and P_{ttd} are forced to converge as snakes. This allows us to find a balance between the expected shape of the contour and the expected location of the contour determined from the images.

We restrict the nodes of the snake to reside on rays emanating from a center point, $\mathbf{x}_s$, which is computed as the centroid of the ray centers, $\mathbf{x}_{f,n}$, where $n \in \{1, ..., m\}$. The rays are equally spaced at an angular offset of $\Delta\phi$ (Fig. 2.8). For snake n, the radial location of the snake node on ray ϕ_τ is referred to as $\mathrm{R}_n(\phi_\tau)$, where $\tau \in \{1, ..., T\}$ and T is the number of rays, and is initially determined from the location of C_n.

Snakes evolve simultaneously such that they are forced to converge to a single snake. The energy used in the snake evolution is

$$\mathrm{E}_{snake,n} = \omega_{drv}\mathrm{E}_{drv,n} + \omega_{xy}\mathrm{E}_{xy,n} + \omega_z\mathrm{E}_{z,n} + \omega_{att}\mathrm{E}_{att} \tag{2.12}$$

where ω_{drv}, ω_{xy}, ω_z, and ω_{att} are scalar weights. $\mathrm{E}_{drv,n}$ is an image energy derived from the projection images, $\mathrm{E}_{xy,n}$ is the curvature energy of the snake in the projection plane, $\mathrm{E}_{z,n}$ is the curvature energy of MV_{surf} at the snake location, and E_{att} is an attractive energy which forces the multiple snakes to converge to a single snake.

Image Energy

The image energy, $\mathrm{E}_{drv,n}$, is derived in much the same way that $\mathrm{F}_{drv,n}$ is derived for the graph. P_{int} and P_{ttd} are first filtered using a Gaussian, $G(\sigma_s)$, and normalized to the range of [0-1] to generate $\mathrm{F}_{int,s}$ and $\mathrm{F}_{ttd,s}$, respectively. The image energy is then

$$\mathrm{E}_{drv,n} = (1-\mathrm{F}_{int,s})^2 \left(\lambda_{ttd}\frac{\partial \mathrm{F}_{ttd,s}}{\partial R} - \lambda_{int}\frac{\partial \mathrm{F}_{int,s}}{\partial R}\right)\Bigg|_{\mathrm{R}_n} \tag{2.13}$$

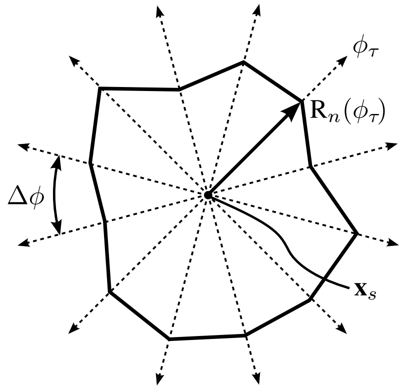


Figure 2.8: Ray system on which snake nodes are forced to reside during contour refinement and convergence, shown with an example snake

where λ_{int} and λ_{ttd} are the same scalar weights used in (2.9), and $E_{drv,n}$ is normalized so that its magnitude is less than one. Imparting force $U_{drv,n}$ defined as

$$U_{drv,n} = -\frac{\partial E_{drv,n}}{\partial R} \tag{2.14}$$

on the snake nodes subsequently minimizes the image energy at the location of the snake.

Contour Curvature Energy

To account for contour curvature but avoid undesirable evolution of the snake in the absence of strong image forces, as is common with typical methods [106], we use the method presented in [102]. This method, instead of minimizing the curvature of the snake, minimizes the change in curvature. The energy $E_{xy,n}$ is then

$$E_{xy,n} = \left| \frac{\partial \kappa_n}{\partial s} \right| \tag{2.15}$$

where κ is the curvature of the contour and s is arc length.

Whereas the method in [102] minimizes (2.15) by moving snake nodes to the perpendicular bisector of neighboring nodes, we restrict the nodes to move along the rays. In doing so, we find $R_{n,des}(\phi_i)$, which is the desired location of the node from snake n along ray ϕ_i which will minimize (2.15). Snake nodes are forced toward the location of $\mathrm{R}_{n,des}(\phi_i)$ using $\mathrm{U}_{xy,n}(\phi_i) = \mathrm{R}_{n,des}(\phi_i) - \mathrm{R}_n(\phi_i)$.

Surface Height Curvature Energy

The shape of the surface at a snake's location is easily obtained, as it is known that MV_{surf} can be described as the graph of a function, Z_{surf}, which is defined in the projection space. Therefore, we can define $\mathrm{Z}_n(\phi_\tau)$ to be the height of MV_{surf} above the node on snake n residing on ray ϕ_τ. Since the annulus is not flat, but rather saddle-shaped, $\mathrm{E}_{z,n}$ is defined as

$$\mathrm{E}_{z,n} = \left| \frac{\partial^2 \mathrm{Z}_n}{\partial \phi^2} \right|. \tag{2.16}$$

This energy is minimized by imparting a force

$$\mathrm{U}_{z,n} = -\frac{\partial \mathrm{E}_{z,n}}{\partial R} \tag{2.17}$$

on the nodes, which encourages a constant local change in elevation of the annulus contour with respect to changes in ϕ.

Attractive Energy

As a means to allow the snakes to interact - so that a node of one snake which is at the global minimum can draw nodes of other snakes away from local minimums - and consequently force the snakes to converge to a common location, we designed the attractive energy E_{att}

$$\mathrm{E}_{att}\left(\phi_\tau\right) = \mathrm{R}_{n_{min}}\left(\phi_\tau\right) - \mathrm{R}_{n_{max}}\left(\phi_\tau\right) \tag{2.18}$$

$$n_{min}\left(\phi_\tau\right) = \underset{n}{\operatorname{argmin}}\left(\mathrm{R}_n\left(\phi_\tau\right)\right)$$
$$n_{max}\left(\phi_\tau\right) = \underset{n}{\operatorname{argmax}}\left(\mathrm{R}_n\left(\phi_\tau\right)\right).$$

E_{att} is minimized in each ray by driving the nodes from the different snakes on that ray towards the center of the node range, $\mathrm{R}_{mid}\left(\phi_{\mathcal{T}}\right) = \frac{1}{2}\left(\mathrm{R}_{n_{min}}\left(\phi_{\mathcal{T}}\right) + \mathrm{R}_{n_{max}}\left(\phi_{\mathcal{T}}\right)\right)$, using $\mathrm{U}_{att,n} = \mathrm{R}_{mid} - \mathrm{R}_{n}$.

Snake Update

Given the described energies which evolve the individual snakes, and the methods designed to minimize the energies, the snakes update according to

$$\mathrm{R}_{n,t+1}{=}\mathrm{R}_{n,t} + \mathrm{d}t\left(\omega_{drv}\mathrm{U}_{drv,n}{+}\omega_{xy}\mathrm{U}_{xy,n}{+}\omega_{z}\mathrm{U}_{z,n}{+}\omega_{att}\mathrm{U}_{att,n}\right) \tag{2.19}$$

where $\mathrm{R}_{n,t+1}$ is the updated snake, $\mathrm{R}_{n,t}$ is the current snake, and dt is a time step. The snakes are updated until they stabilize to within 0.1 pixels, which occurs in roughly 100 iterations using the parameter values shown in Table 2.1.

2.1.8 3D Mitral Annulus Contour

The three-dimensional annulus is constructed from the final snake by computing the $\mathbf{x}'$ and $\mathbf{y}'$ coordinates using the angle of the rays and radial location of the nodes, and the $\mathbf{z}'$ coordinates using Z_n. We compute a continuous annulus contour in $\Re^3$ using a cubic interpolation of the defined points.

2.1.9 Algorithm Convergence

The point $\mathbf{x}_c$ which initially positions and orients the graph used to find MV_{surf} is provided initially by a user who is asked to provide a point near the center of the valve. Therefore, the location of the user-specified point could have an effect on the shape of the 3D annulus contour. However, this dependency is resolved by iterating the algorithm until convergence metrics are satisfied (Fig. 2.9).

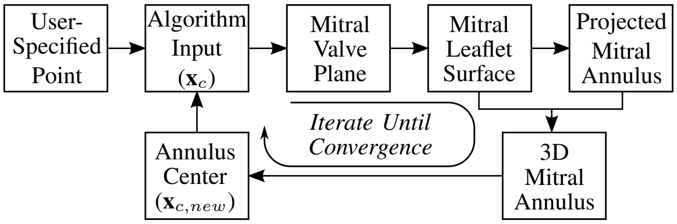


Figure 2.9: Simplified flow chart for the mitral annulus segmentation algorithm; the algorithm iterates until the computed annulus center stops changing.

After the first execution of the algorithm, and in subsequent iterations, the center point of the resulting 3D annulus contour, $\mathbf{x}_{c,new}$, is compared to the location of $\mathbf{x}_c$. If $|\mathbf{x}_{c,new} - \mathbf{x}_c| \quad > \quad \epsilon$, where ϵ is a pre-defined scalar distance, $\mathbf{x}_c$ is set equal to $\mathbf{x}_{c,new}$ and the algorithm iterates. Otherwise, it is assumed the annulus contour has been accurately segmented. Enforcing this convergence means the resulting annulus segmentation is independent of the initial user-provided point, making the algorithm's results operator-independent. Section 2.2.3 shows that a relatively large region of convergence exists.

2.2 Performance and Validation

To determine the appropriateness of the algorithm for diagnostic applications, we assessed the algorithm's abilities to accurately locate the mitral annulus in 3DUS compared to a group of experts (Sec. 2.2.1). For surgical planning purposes, we compared the algorithm's results to the appearance of the mitral annulus as seen in a surgical view (Sec. 2.2.2). We also explored the sensitivity of the algorithm to the initial user-specified point which serves as the algorithm's input (Sec. 2.2.3).

It is important to note that in these studies, the same parameter values (Table 2.1) were used for every valve and every study. These parameters were found by testing the algorithm on 11 clinical images, none of which were included in the validation studies. The clinical images contained both normal and diseased mitral valves (3 normal, 4 with

Table 2.1: Mitral Annulus Segmentation Algorithm Parameters Used for All Studied Valves

Parameter	Value	Units	Parameter	Value	Units
σ	1	mm	α_c	25	
p	3	voxels	m	2	
r_{pca}	12.5	mm	$\sigma_{n=1}$	2.5	mm
r_{graph}	25	mm	$\sigma_{n=2}$	1	mm
ω_p	1.5		$\Delta\phi$	10	degrees
ω_o	1		σ_s	1	mm
α_s	25		$\mathrm{d}t$	0.2	
$\Delta\gamma$	5	degrees	ω_{drv}	2	
Δr	0.5	mm	ω_{xy}	2	
λ_{int}	5		ω_z	0.2	
λ_{ttd}	1		ω_{att}	1	
ω_s	25		ϵ	0.5	mm
ω_d	1				

mitral prolapse, and 4 with mitral regurgitation). Of the 11 images, 10 were acquired with a transesophageal approach (X7-2t probe), and one with a transthoracic approach (X7-2 probe).

The algorithm was coded mostly in MATLAB (The MathWorks, Natick, MA, USA), but used C++ for both computing Φ in the TTD, and for the implementation of the max-flow algorithm. Approximate running times for various portions of the algorithm (64-bit PC, 3.0 GHz Intel Core 2 Duo processor with 4GB of RAM) were 30 seconds to compute the TTD, 30 seconds to compute the mitral leaflet surface, and 15 seconds to subsequently initialize and converge the contours, meaning each iteration took roughly 45 seconds. Typically two iterations were needed for convergence.

2.2.1 Validation Study Using Manual Image Delineations

Noise in 3DUS and volume visualization limitations make delineating the annulus from 3DUS difficult. As a result, a ground truth segmentation of the annulus cannot be defined from images. Therefore, to validate the annulus segmentation algorithm, we compared the algorithmic annulus to manual segmentations of the annulus performed by a group

Table 2.2: Summary of Clinical Mitral Valve Images Used in Manual Image Delineation Validation Study

Valve	Mitral Valve State	Acquisition Type	Probe
1	Prolapse; Regurgitation	Transesophageal	X7-2t
2	Normal	Transesophageal	X7-2t
3	Normal	Transthoracic	X7-2
4	Normal	Transesophageal	X7-2t
5	Cleft Mitral	Transesophageal	X7-2t
6	Prolapse; Dilated Annulus	Transesophageal	X7-2t
7	Prolapse; Myxomatous Valve	Transesophageal	X7-2t
8	Prolapse; Regurgitation	Transesophageal	X7-2t
9	Normal	Transesophageal	X7-2t
10	Prolapse	Transthoracic	X7-2

($n = 10$) of cardiologists and trained echocardiography technicians, who will be collectively referred to as experts. We performed the analysis by first comparing the segmentations made by each expert to the collection of segmentations made by the rest of the experts for each valve. In this way, a measure of the performance for each expert could be quantified. We then compared the algorithm segmentations to the collection of segmentations made by the entire group of experts for each valve.

Experts were provided with 3DUS images of 10 different closed valves. The state of the valve and the nature of the 3DUS acquisition for each valve is summarized in Table 2.2. The images were slices at 10^o increments about a mitral valve center point and axis. We asked the experts to delineate the two annulus points in each image of the valve, for a total of 36 annulus points per valve. While the segmentations were performed in a single frame, temporal information in each image was available to the experts to allow them to accurately delineate the location of the annulus. To reduce the effects of training and fatigue in the analysis, we presented the valves to the experts in random order.

The comparison of an expert to other experts, or the algorithm to the group of experts, was done on a point-by-point basis. As some annulus points were better defined than others, it was appropriate to penalize more for deviations of a point from a well-

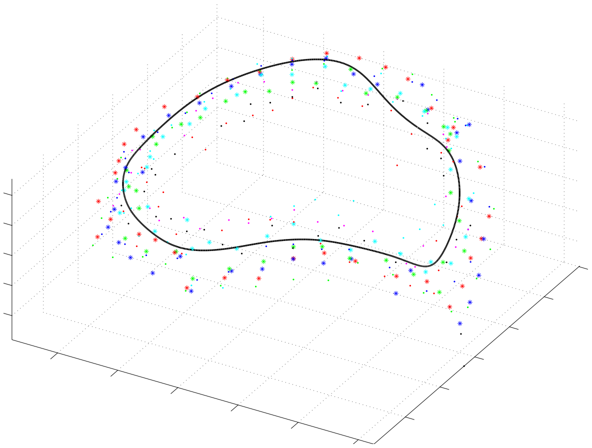


Figure 2.10: Typical comparison between the algorithmic annulus *(solid line)* and points delineated manually by experts on clinical 3DUS images

defined annulus location than from a poorly-defined location. The measure of a well-defined and poorly-defined annulus location was based on the collection of segmentations to which a particular segmentation was being compared. Additionally, as the experts could only provide points in the space of the image, points did not deviate out of the image plane. Therefore, the analysis of points was done in $\Re^2$ space.

When comparing an expert's segmentation to the collection of segmentations made by the remaining experts, we refer to the i^{th} point of the expert's segmentation as $\mathbf{A}_i$, where $i \in \{1, ..., 36\}$. The mean and covariance of the i^{th} point as determined from the segmentations from the rest of the group are referred to as $\overline{\mathbf{A}}_i$ and $\mathbf{\Sigma}_i$, respectively. A normalized (Mahalanobis) distance was then computed for each $\mathbf{A}_i$, and the performance of an expert quantified as

$$MND = \frac{1}{36} \sum_{i=1}^{36} \sqrt{\left(\mathbf{A}_i - \overline{\mathbf{A}}_i\right)^{\mathrm{T}} \mathbf{\Sigma}_i^{-1} \left(\mathbf{A}_i - \overline{\mathbf{A}}_i\right)} \tag{2.20}$$

which is the mean normalized distance (MND) across all points for a valve.

We then analyzed the algorithm results in much the same way we analyzed the segmentations from each expert, where the i^{th} point of the algorithm's segmentation was

Table 2.3: Mean Normalized Distances & Distribution Parameters for the Experts and Algorithm

	Valve										Mean ±
Expert	1	2	3	4	5	6	7	8	9	10	Std. Dev.
1	1.23	1.19	1.06	1.04	1.16	1.74	1.07	0.94	1.28	1.00	1.63±0.76
2	1.19	1.31	1.98	1.32	1.64	1.82	1.02	1.46	1.38	2.63	
3	2.68	1.50	1.69	2.68	1.92	1.62	2.71	2.61	1.85	3.80	
4	2.19	1.74	3.05	0.83	1.56	2.03	2.32	1.02	1.09	0.86	
5	1.50	1.09	1.39	1.25	1.21	0.99	1.10	0.91	1.94	0.92	
6	1.03	1.79	1.19	0.88	1.65	1.10	0.96	0.95	0.81	0.93	
7	1.03	1.15	1.07	1.14	1.33	1.08	1.69	0.94	1.42	1.26	
8	1.15	1.71	1.25	1.18	1.51	0.96	1.19	2.31	2.36	1.13	
9	1.44	1.73	1.40	4.16	1.01	1.23	1.49	1.36	1.34	0.98	
10	2.70	2.97	1.92	2.00	2.97	4.33	2.81	3.66	2.46	2.92	
Algorithm	1.19	0.97	1.42	0.97	1.15	1.13	1.28	1.03	1.19	0.73	1.11±0.19

$\mathbf{A}_i$, and $\overline{\mathbf{A}}_i$ and $\mathbf{\Sigma}_i$ were the mean and covariance, respectively, of the i^{th} point from the entire group of experts. The *MND* computation for the algorithm was then the same as shown in (2.20).

A typical comparison between the algorithm's resulting annulus and the points manually delineated by the experts can be seen in Fig. 2.10. Table 2.3 shows the *MND* values for each expert and for the algorithm across all valves, along with the sample mean and standard deviation of the *MND* values from each group. Based on the distribution parameters of the two groups, we concluded that the algorithm (*MND*$= 1.11 \pm 0.19$) had a lower mean *MND* than the group of clinicians (*MND*$= 1.63 \pm 0.76$), and that this difference was statistically significant ($p < 0.001$). If instead of looking at the group, we look at the best scoring expert (Expert 6: *MND*$= 1.13 \pm 0.33$), we found that the algorithm and the best scoring expert were statistically indistinguishable ($p = 0.85$).

We also compared the algorithm and experts by computing the RMS difference between the algorithm and the expert average (Table 2.4), as this is a commonly used accuracy metric and allows for comparisons to be drawn to other studies and methods. The average RMS difference between the algorithm and the expert average was 1.81 ± 0.78mm.

Table 2.4: RMS Difference Between the Algorithmic Annulus and Manual Delineations Made by Experts on Clinical Images

Valve	RMS Difference (mm)
1	1.95
2	1.37
3	2.75
4	0.96
5	1.19
6	2.50
7	3.22
8	1.82
9	1.27
10	1.09
Mean±Std. Dev.	1.81±0.78

The larger RMS differences usually coincided with a disagreement among the experts as to the annulus location (i.e. a large spread of experts' annular points), which could typically be attributed to poor image quality.

2.2.2 Validation Study Using Surgical View Delineations

Given the poor image quality of 3DUS, validating the algorithm by comparing the algorithm's annulus to manual delineations made in 3DUS only answered the question of whether the algorithm was interpreting the images in the same way as a group of experts. The conclusion could not be made, however, that the annulus was understood to be the same shape as seen in a surgical view. Therefore, we further validated the algorithm by comparing the algorithm results to a fully visible mitral annulus delineated by a pointing device tracked using an electromagnetic (EM) tracker (miniBIRD - Model 800, Ascension Technology Corporation, Burlington, VT, USA).

We performed this validation study using six freshly excised porcine hearts. We removed the left atrium to fully expose the mitral annulus, and to inhibit motion, secured the heart to a plastic frame using suture. After securing the heart in a water tank, to roughly simulate a peak systolic state, we artificially distended the left ventricle by connecting the

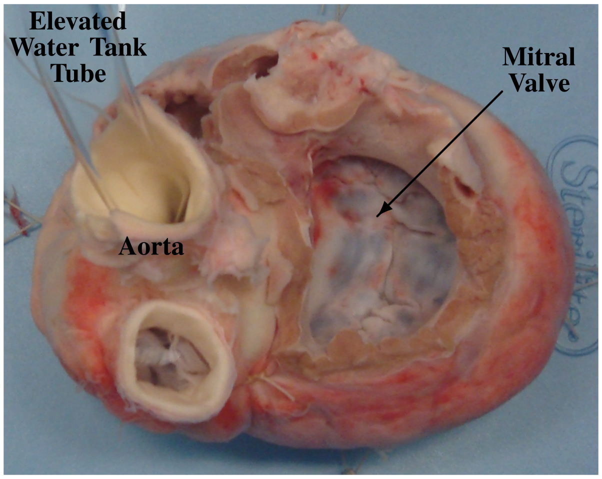


Figure 2.11: Typical view of a loaded mitral valve in a water tank that was used in the surgical view delineation study.

Table 2.5: RMS Difference Between the Algorithmic and EM Delineated Annulus from Freshly Excised Porcine Hearts

Valve	RMS Difference (mm) (average ± std. dev.)
1	1.15 ± 0.04
2	1.35 ± 0.18
3	1.42 ± 0.01
4	1.38 ± 0.12
5	0.95 ± 0.42
6	0.91 ± 0.26
Mean	1.19 ± 0.17

aorta to a tube from a container of water elevated about two meters above the heart, and by closing the coronary arteries using suture (Fig. 2.11). An expert acquired a 3DUS volume of the loaded mitral valve with a transthoracic probe, and shortly after, used an EM tracked pointing device to delineate the fully visible mitral annulus with about 30 evenly placed points. Taking into account the sensor resolution and calibration procedure, point locations were computed to within a measured accuracy of 0.21 mm.

We segmented the annulus from the acquired 3DUS volume using the semi-automatic algorithm, and registered the EM delineated points to the algorithmic annulus by first manually aligning the two segmentations and then using an iterative-closest-point algorithm to

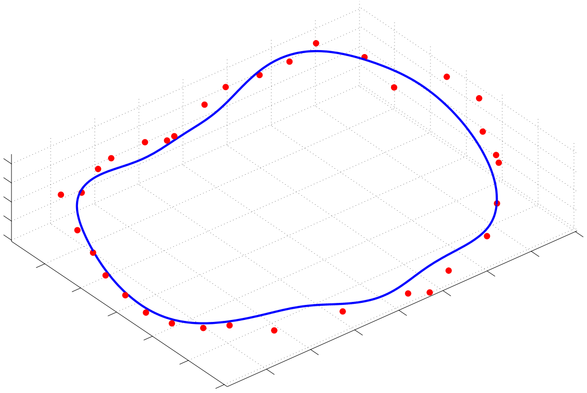


Figure 2.12: Typical comparison between the algorithmic annulus *(solid line)* and points delineated by an expert on a porcine mitral annulus using an EM tracked device.

refine the alignment. The RMS differences between the two can be seen in Table 2.5, with Fig. 2.12 showing a typical comparison. The average RMS distance of the EM delineated annulus from the algorithmic annulus was 1.19 ± 0.17mm, where we computed the standard deviation of the RMS distance from three different EM delineations of the annulus made by the expert on each mitral valve.

2.2.3 Sensitivity Study

Just as important as the algorithm's accuracy is the accuracy needed for the algorithm's input, which is the user-specified point. For this study, we first established a baseline segmentation for each valve, i.e. the algorithm's segmentation resulting from a carefully placed user-specified point near the center of the valve. The algorithm was then run on test points that were placed in the region around the center point, where we determined the center point from the baseline segmentation. If we refer to a segmentation resulting from a test point as a test annulus, then a test annulus was said to converge to the baseline if all points from the test annulus were less than 0.5 millimeters away from the baseline. The 3DUS images used in this study were the same as those used in the imaging validation study described in Section 2.2.1.

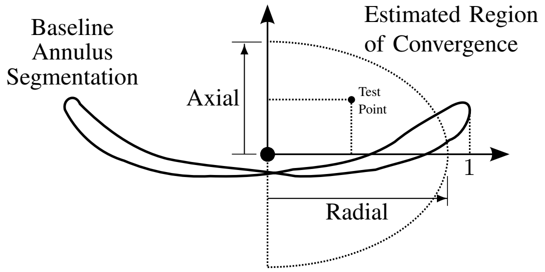


Figure 2.13: Estimated shape and size of the region of convergence relative to a baseline segmentation and described using the axial and radial directions

Table 2.6: Estimated Region of Convergence Dimensions for Each Valve Normalized by the Annular Radius

Valve	Axial	Radial
1	0.96	1.34
2	0.44	0.65
3	0.76	1.13
4	1.24	1.08
5	0.72	1.11
6	0.71	0.98
7	0.96	1.16
8	0.78	0.68
9	1.26	1.17
10	0.67	1.26
Mean ± Std. Dev.	0.85 ± 0.26	1.06 ± 0.23

Based on the test points that resulted in the algorithm converging to the baseline, we could then establish a region of convergence. To describe the region of convergence in general terms, we refer to two directions – an axial direction, which is an estimation of the mitral valve axis from the baseline segmentation, and a radial direction, which is any direction perpendicular to the axis extending out from the center point towards the annulus. We represent the location of all test points relative to these axes, and normalize the coordinates by the radius of the baseline annulus in the plane defined by the axis and the test point. We then approximate the region of convergence as the ellipsoidal region shown in Fig. 2.13. Table 2.6 shows the results of the sensitivity study, where it can be seen that the region of convergence extends to an average of 85% of the annulus radius in the axial direction and 106% of the annulus radius in the radial direction.

2.3 Discussion

2.3.1 Performance and Validation

In both validation studies, the mitral annulus segmentation algorithm accurately and robustly segmented the three-dimensional mitral annulus contour from 3DUS. When comparing the algorithm to a group of experts, results showed that the algorithm consistently segmented the annulus near the location selected by the experts (Table 2.3). The consistency of the algorithm was better than the group of experts, i.e. while the algorithm might not have the lowest *MND* values for each valve, across all valves, the algorithm had a lower mean *MND* than the group of clinicians ($p < 0.001$). Additionally, the algorithm and the best scoring expert were statistically indistinguishable ($p = 0.85$). We also analyzed the expert segmentations by computing the RMS difference between the algorithm and the expert average, in which we found an average RMS difference of 1.81 ± 0.78mm. Lastly, we compared the computed annulus shape to a fully exposed porcine annulus delineated using an EM tracked pointer. Results showed an average RMS difference of 1.19 ± 0.17mm between the algorithm and the delineated points. This smaller RMS difference, compared to the validation study using the group of experts, is due to the fact that the true annulus could be seen. In contrast, the group of experts could only use 3DUS images to interpret the annulus location. Consequently, when image quality was poor, there was typically a lack of consensus as to the annulus location, resulting in a larger RMS difference between the expert average and the algorithm.

The two validation studies were complimentary. The first study used actual clinical images, which display the full range of noise and distortion encountered in medical practice. Here the performance of the algorithm cannot be compared to a “gold standard” determination of the annulus location; instead, the results were compared to the annulus location determined by a group of practicing clinicians. This approach is most immediately relevant to diagnostic applications, where determination of the annulus shape provides insight into various pathological states [51, 21, 41, 40, 72]; current clinical practice is based on the expert segmentations which are the standard for comparison here.

In the water tank study, the quality of the 3DUS images is significantly better than in vivo, which may improve algorithm performance compared to clinical images. The annulus location, however, could be directly designated using visual inspection and a pointing device, which is likely to produce a good estimate of the "true" annulus location. This method is particularly pertinent to surgical planning applications [19], as the mitral annulus is viewed from a similar perspective during annuloplasty. While both validation studies have limitations, the results in combination suggests that the algorithm is able to estimate annulus location with an accuracy comparable to current clinical practice.

The sensitivity study suggests that the algorithm is robust to variation in user input. So long as the user-specified point is located within about a half-diameter of the valve's center, the algorithm converges to the same annulus location. Given that this point is the only information the algorithm requires from the user, the results of the semi-automatic annulus segmentation algorithm are consequently independent of the user as well. Additionally, despite variation in the states and degrees of pathology of the mitral valves, the probe orientation relative to the valve, and the ultrasound machine settings, the algorithm accurately segmented the annulus using the same algorithm parameter values for every image, reinforcing the robustness of the method.

The measured accuracy of this algorithm is difficult to compare to prior work, as previously published semi-automatic mitral annulus segmentation algorithms include minimal validation [107, 99]. In [99], indirect quantitative analysis of the method was performed by comparing measurements from a model to published population valve dimensions. As an example, for average annulus diameter dimensions, the method reported errors of roughly 15%. We measured the annuli from our algorithm and the EM delineations discussed in 2.2.2 and found an average diameter difference of under 4%.

2.3.2 Algorithm Design

The algorithm uses a thin tissue detector which can be likened to a simplified three dimensional ridge detector [55] for determining the valve orientation, computing the mitral leaflet surface, and finding the location of the projected mitral annulus contour. This is a key reason the algorithm is successful: the only extensive regions of thin tissue in the image volume are the mitral valve leaflets, so the algorithm focuses on the mitral apparatus and ignores the rest of the volume. This generates accurate results despite poor placement of the user input point.

Characterizing the leaflet location using the thin tissue detector, however, does place restrictions on the ultrasound imaging, in that the thin leaflet tissue needs to be visible (Fig. 2.14(a)). This is not the case when the ultrasound wave propagation is parallel to the leaflet surface orientation (Fig. 2.14(b) and 2.14(c)), as this produces little acoustic return. The algorithm will fail on images such as these. However, similar restrictions would also exist for the previous work in annulus segmentation presented in [107] and [99], and is to be expected for algorithms wishing to delineate the mitral annulus from a single ultrasound frame and with no use of temporal information.

With ultrasound imaging, anatomical structures become more or less defined depending on their orientation to the acoustic propagation (Fig. 2.14). This helps to explain why the presented method cannot be used to find the annulus throughout the cardiac cycle, but rather was designed for just closed valves. For instance, if the probe orientation such as that shown in Fig. 2.14(a) was used to view the mitral valve throughout the cardiac cycle,

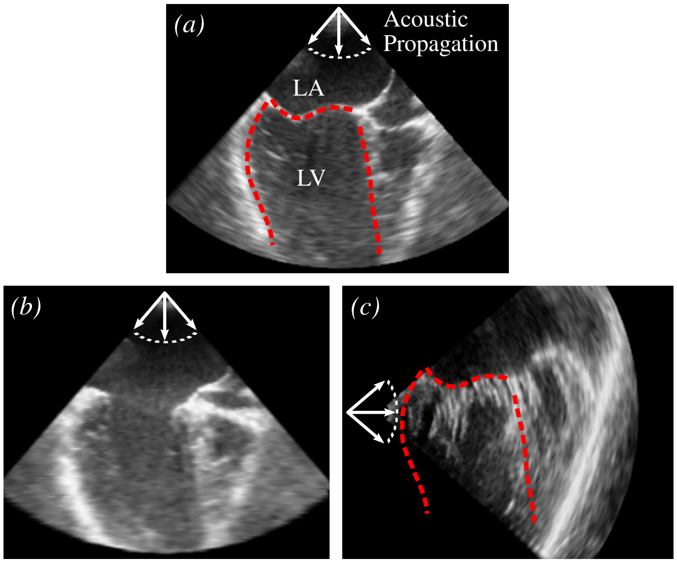


Figure 2.14: Mitral valve leaflet appearance of the same valve in several scenarios. *(a)* Closed mitral valve in a long-axis view (LA - left atrium; LV - left ventricle). Leaflet surface is well defined. Dashed red lines are the approximate border of the LV. *(b)* Open mitral valve in a long-axis view. Leaflets are poorly defined. *(c)* Closed mitral valve in a short-axis view shown with the border taken from *(a)*. Leaflet surface is poorly defined and chordae insertions are visible.

when the valve opens (Fig. 2.14(b)), the leaflets become oriented such that they become poorly defined.

The graph cut method (max-flow algorithm) is used in the annulus segmentation algorithm first in a 3D graph to find a surface at the location of the mitral leaflets, and then in a 2D graph to find a contour at the location of the projected mitral annulus in the projection images. There are several reasons why we use the graph cut method. One of the main reasons is the fact that it finds an optimal solution to an energy minimization problem, meaning we will not fall victim to the local minima that many methods are subject to which attempt to find an optimal surface or contour (i.e. active contours[43], level sets[84], etc.). This is especially important in cardiac ultrasound, where anatomic variation along with the noise, artifacts, and inhomogeneities inherent in ultrasound imaging have the potential to create several local minima. Other reasons to use the graph cut method are that it does not

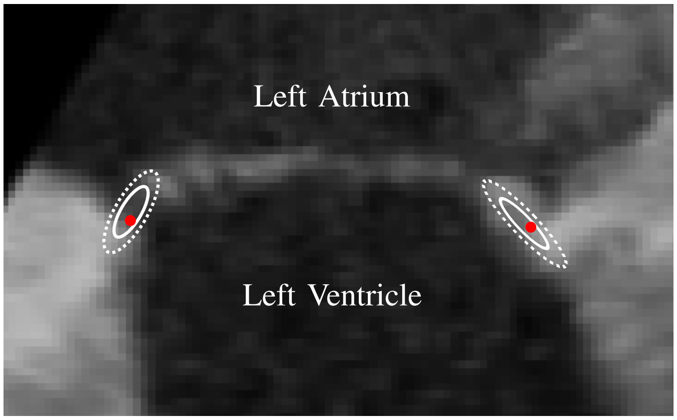


Figure 2.15: Slice from 3DUS of the mitral valve used in the manual segmentation study showing typical distributions of the annulus points selected by the experts. The first standard deviation *(solid line)* and second standard deviation *(dotted line)* are shown. The red point is the location of the algorithm-generated annulus in the slice. This suggests that human observers, much like the algorithm, select annulus points along the leaflet structure.

require an initialization, and a limited amount of prior knowledge can be enforced through mindful construction of the graph. Therefore, the need for additional user input, for either initialization or to encourage a specific geometry, is eliminated without sacrificing accuracy, and we avoid the variation inherent in any additional user inputs. The combination of these characteristics of the graph cut method helps to make the annulus segmentation algorithm accurate, and also helps to explain why the algorithm has such a large region of convergence relative to the user-provided point.

The annulus segmentation algorithm can be simplified to two main steps: find the leaflets, then search along the leaflets to find the annulus. This was consistent with the approach used by the experts (cardiologists and technicians) to delineate the annulus. Fig. 2.15 shows an example of the distribution of the human selected annular points which are aligned along the leaflets. The main difference was that after finding the leaflets, the human observers had access to temporal information to delineate the annulus location, whereas the algorithm used only a single 3DUS volume.

2.3.3 Future Work

One of the most significant improvements to the algorithm would be to integrate temporal information to more accurately and robustly locate the annulus. Currently, the algorithm finds the annulus as the location at the leaflets where the tissue thickness changes abruptly along the computed leaflet surface. A drawback of this approach, however, is that if the TTD is improperly tuned or if the leaflets are severely calcified, the annulus might not be accurately located, which might be ameliorated when temporal information is used.

The large region of convergence for the algorithm makes it an attractive candidate for complete automation. Because the only input to the algorithm is the user-specified point, if this point can be delineated by some automated process, the annulus segmentation algorithm could be made fully automatic. Special care would need to be taken, however, to account for the different appearances of valves due to varying pathologies, ultrasound probe orientations, and ultrasound machine settings. An example of a method that could be used for this automatic selection is the work presented in [97], which automatically segments the mitral valve plane and left ventricular long-axis. The intersection of the plane and the axis could be used to define the user-specified point.

The annulus segmentation algorithm presents many opportunities in clinical studies. The algorithm can help to analyze large clinical databases to explore correlations between pathologies and annular geometries. It can also be used to monitor patient anatomy over time to determine if geometry at an earlier age has any indication as to pathologies that may develop at later ages.

Given that the algorithm can segment the mitral annulus whenever the leaflets are coapted, the shape and motion of the annulus throughout most of the cardiac cycle can already be obtained from the presented method. We show in the next chapter how to use this method to find the mitral annulus throughout an entire cardiac cycle.

Chapter 3

Four-Dimensional Mitral Annulus Segmentation and Tracking

Methods to delineate annular shape and motion include tracking implanted radiopaque markers with fluoroscopy [25, 13, 93, 6], tracking implanted sonomicrometry transducers [27], and time consuming manual segmentation from imaging [51, 35, 41, 21, 1, 40, 72, 14, 2, 74, 80, 104]. Few automated methods have been developed to delineate its four-dimensional (4D) shape from ultrasound. There have been a number of semi-automatic methods to track the annulus in 2D ultrasound that generally require manual initialization of the mitral anatomy, after which the annulus points are tracked using some form of either block or template matching, optical flow, active contours, dynamic programming, or a combination of these methods [18, 91, 58, 66]. One of the first methods to track the annulus in 4DUS is the work by Veronesi *et al.*, 2006 [98], which uses a combination of optical flow and block matching to track manually initialized annulus points in 4D. This method, however, does not constrain the motion of the annulus, leading the authors to add that "manual corrections were performed when required," revealing that the method

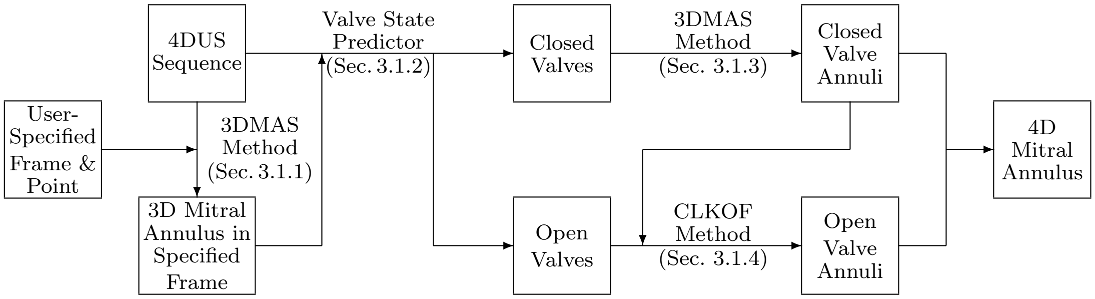

4DUS Sequence
User-Specified Frame & Point
3DMAS Method (Sec. 3.1.1)
3D Mitral Annulus in Specified Frame
Valve State Predictor (Sec. 3.1.2)
Closed Valves
Open Valves
3DMAS Method (Sec. 3.1.3)
CLKOF Method (Sec. 3.1.4)
Closed Valve Annuli
Open Valve Annuli
4D Mitral Annulus

Figure 3.1: Flow chart outlining the new 4D mitral annulus segmentation (4DMAS) algorithm. Specific details for the respective processes can be found in the indicated sections.

accumulates tracking error that causes the delineations to become inaccurate. Ionasec *et al.*, 2010 [37] delineate the 4D annulus by fitting a mitral valve model to 4DUS data using machine learning techniques. The method, while accurate, requires an expansive database of thousands of manually delineated features (that took the authors two years to compile), making the method inaccessible to most.

Current 4D annulus delineation methods either require extensive user interaction, do not take into account the shape or motion of the annulus, or accumulate excessive tracking error. To address these issues, we present a new semi-automated 4D mitral annulus segmentation (4DMAS) method for delineating the annulus in a 4DUS sequence (Figure 3.1). From the user, it requires only the selection of one frame with a closed valve and one point near the valve. Using these inputs and the 3D mitral annulus segmentation (3DMAS) method for closed mitral valves presented in Chapter 2, the annulus in the selected frame is found. A valve state predictor then uses this annulus to determine those frames that contain a closed valve versus those that contain an open valve. For those containing a closed valve, the 3DMAS method is again used. For those containing an open valve, a constrained optical flow formulation based on the Lucas & Kanade method [57] is used to track the annulus. In taking into account both the valve state and the shape and motion of the annulus in the optical flow method, the 4DMAS method limits the accumulation of segmentation error.

The details of the 4DMAS method are presented in Section 3.1. An analysis of the 4DMAS method is presented in Section 3.2, which includes studies that examine the accuracy of the valve state predictor (Section 3.2.1) and the accuracy of the constrained optical flow method (Section 3.2.2).

3.1 Materials and Methods

3.1.1 3D Mitral Annulus Segmentation Algorithm

The presented 4DMAS method builds on our previously presented 3DMAS algorithm, which is able to accurately segment the location of the annulus for closed mitral valves. It requires only the selection of a frame with a closed valve and a point near the center of the valve. The 3DMAS method ultimately provides the geometric prior of the annulus which we then track to frames with an open valve. The methods of computing a thin tissue detector and finding a leaflet surface, which were developed for the 3DMAS algorithm, are also used in the 4DMAS method to help determine the valve state throughout a sequence. The following sections explain how to integrate the 3DMAS method and its components into a 4D annulus tracking method that first predicts the valve state (open or closed) and then alters the segmentation approach based on valve state so that an accurate 4D annulus results with little accumulation of error due to tracking drift.

3.1.2 Valve State Predictor

The 4DMAS method begins by first implementing the 3DMAS method on a user-specified frame using a specified point. The selected frame will be referred to as the reference frame, F_{ref}, and the resulting annulus as the reference annulus, A_{ref}. Predicting the valve state in all other frames is then done using information gathered from this segmentation. To begin, a mitral leaflet surface is computed in each frame as described in Chapter 2.

Regardless of whether the valve is open or closed in a frame or whether the leaflets are visible or not, due to the nature of the surface generation method, which is a max-flow algorithm run on a cylindrically shaped graph surrounding the valve, a surface is always found in each frame (Figure 3.2). When the valve is closed, the surface will reside at the leaflets, however, when the valve is open, the surface will reside somewhere in the blood pool formed by the left atrium and left ventricle. We then assume the value of the thin tissue detector and ultrasound intensity at the surface to be higher for closed valves than for open valves, and based on this we differentiate between the two states. Because we are not interested in the surface outside of the region of the valve, and because we know the valve moves roughly along an axis that we can estimate from A_{ref} [44], we crop the surfaces in every frame by the projection of A_{ref} along the precomputed axis (Figure 3.2), keeping only the region of the surfaces inside of the projected contour. We then compute the mean ultrasound intensity value, $\mu_{int,n}$, and mean thin tissue detector value, $\mu_{ttd,n}$, at the cropped surfaces. A valve score for each frame is then

$$VS_n = (\mu_{int,n})_{[0-1]} \, (\mu_{ttd,n})_{[0-1]} \tag{3.1}$$

where $(\cdot)_{[0-1]}$ indicates that the values for that term are normalized across all frames to the range of 0 to 1, and n indicates the frame number.

To group the frames into those that have a closed valve versus those that have an open valve, the values of VS_n are clustered using a k-means clustering algorithm. Clusters are initialized by assigning all values greater than the mean of VS_n to the “closed valve” cluster, and all others to the “open valve” cluster. This is based on the assumption that closed valves will have a higher mean ultrasound intensity and thin tissue detector value at the cropped surface than open valves. The distance metric used for the k-means algorithm is the absolute value of the distance of each valve score to the mean of the respective clusters. Lastly, the valve score for F_{ref} is always assigned to the “closed valve” cluster to ensure a

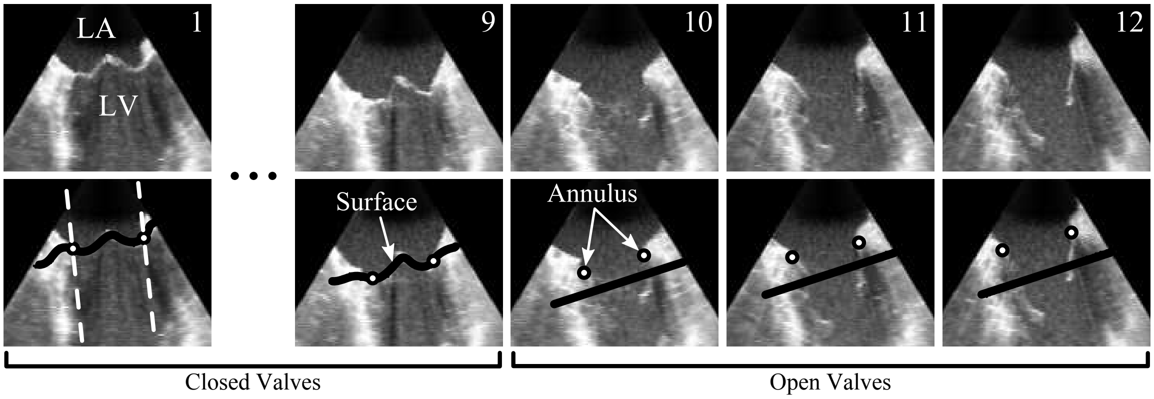


Figure 3.2: (*Top Row*) Slices from a 4DUS sequence showing a prolapsed mitral valve over a cardiac cycle, where the numbers indicate the frame number. (*Bottom Row*) The same slices from the top row shown with the surfaces used in the valve state prediction, and the final location of the segmented annulus. The white dashed lines in the bottom left image indicate the cropping boundaries used in the valve state prediction, and are formed by the projection of A_{ref} along the valve axis. This boundary is constant for all frames. The closed and open valves are those as determined by the valve state predictor. It can be seen that when the valve opens, the surface cannot find closed leaflets and so resides in the blood pool. The annulus is then found by tracking instead of direct segmentation. (LA - Left Atrium; LV - Left Ventricle)

labeling consistent with the user's selection of a closed valve. An example of the clustering results for the ultrasound sequence shown in Figure 3.2 can be seen in Figure 3.3. For simplicity, frames containing a closed valve will herein be referred to as "closed frames," and similarly, frames containing an open valve as "open frames."

3.1.3 Annulus Segmentation for Closed Valves

Knowing those images that contain closed valves via the valve state predictor, we segment the annulus in all closed frames using the 3DMAS algorithm. The initially provided point in each frame that serves as the input to the algorithm is no longer provided by the user, but is determined automatically by projecting the center point of the originally segmented annulus along the valve axis onto the respective mitral leaflet surface that was used to compute the valve score.

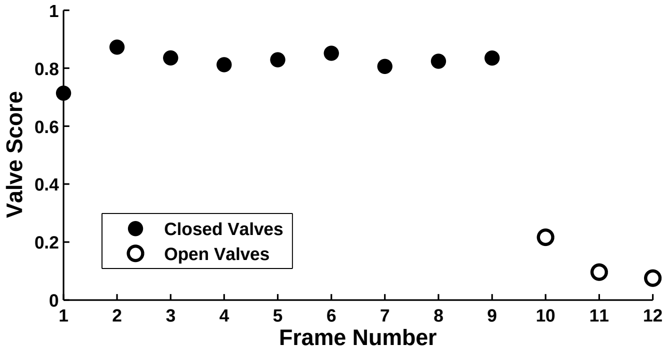


Figure 3.3: Mitral valve scores for the 4DUS sequence shown in Figure 3.2 using Frame 1 as the reference frame.

3.1.4 Constrained Optical Flow

When the valve opens, it is no longer possible to segment the annulus using the 3DMAS method, and we must track the annulus from one frame to the other. Similar to previously published annulus tracking methods, we use a variation of the Lucas & Kanade optical flow (LKOF) algorithm, but unlike previous methods, we incorporate shape information in the formulation. A desirable characteristic of the LKOF method is the ability to compute the optical flow for individual points. This reduces the computational burden that other methods impose, such as the Horn & Schunck method [36], which require the optical flow to be computed over an entire volume. We know the annulus is a continuous anatomical structure, and so we would like to enforce a locally smooth displacement of the annulus. However, unlike the Horn & Schunck method, the LKOF method is not equipped to enforce a locally smooth displacement field. We therefore modify the LKOF formulation to enforce this constraint for the purpose of tracking the mitral annulus.

The LKOF algorithm is based on the assumption that the intensity for a target location changes little from one frame to the next, and is formulated to minimize

$$E = \sum_{x \in R} \left[F\left(x + h\right) - G\left(x\right) \right]^2 \tag{3.2}$$

which is the sum of the squared differences for a small window, R, displaced by h in ultrasound intensity image F relative to image G. We want to track only the annulus from one frame to the next, and so refer to the displacement of point i along the annulus contour as $h_i = [h_{i1}, h_{i2}, h_{i3}]^T$. We know that h_i should exhibit small local deviations, and so in addition to minimizing the change in appearance of a window around a point, we also want to limit its movement based on the movement of neighboring points. We quantify the local deviations in annular displacement as $\partial^2 h_i / \partial s^2$, and therefore aim to minimize at each point

$$E_i = \sum_{x \in R} W(x) \left[F(x+h_i) - G(x)\right]^2 + \alpha \left(\frac{\partial^2 h_i}{\partial s^2}\right)^T \left(\frac{\partial^2 h_i}{\partial s^2}\right) \tag{3.3}$$

with respect to h_i, where α is a scalar weight. We refer to this method throughout the rest of this paper as the constrained Lucas & Kanade optical flow (CLKOF) method. For this formulation, the position along the annulus contour is parameterized by s. In the implementation, we track N evenly spaced points along the contour, making $i \in \{1, ..., N\}$. For this study, we use $N = 24$ and $\alpha = 500$. The weighting function

$$W(x) = \frac{\|\nabla \Omega(x)\|}{\sum_{x \in R} \|\nabla \Omega(x)\|} \tag{3.4}$$

is designed to give more importance to regions located at a high gradient magnitude, where Ω refers to the intensity of the ultrasound volume.

To find the h_i that minimizes (3.3), similar to approaches taken in [57, 43], we first approximate the first order derivative of F and second order derivative of h as

$$\frac{\partial}{\partial x} F(x) \approx \frac{F(x+h_i) - F(x)}{h_i} \tag{3.5}$$

$$\frac{\partial^2 h_i}{\partial s^2} \approx \frac{h_{i-1} - 2h_i + h_{i+1}}{\Delta s^2} \tag{3.6}$$

where $\frac{\partial}{\partial x} = \left[\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \frac{\partial}{\partial x_3}\right]$. We then substitute (3.5) and (3.6) into (3.3), making (3.7) the function to minimize. Setting $\partial E_i / \partial h_i = 0$ and solving for h_i, we get (3.8) as the solution for h_i, where I_3 is the 3×3 identity matrix.

$$E_i = \sum_{x \in R} W(x) \left[F(x) + h_i \frac{\partial}{\partial x} F(x) - G(x) \right]^2 + \alpha \left(\frac{h_{i-1} - 2h_i + h_{i+1}}{\Delta s^2} \right)^T \left(\frac{h_{i-1} - 2h_i + h_{i+1}}{\Delta s^2} \right) \quad (3.7)$$

$$h_i = \left[\sum_{x \in R} \left(\frac{\partial}{\partial x} F(x) \right)^T W(x) (G(x) - F(x)) + \frac{2\alpha}{\Delta s^4} (h_{i-1} + h_{i+1}) \right] \times \left[\sum_{x \in R} \left(\frac{\partial}{\partial x} F(x) \right)^T W(x) \left(\frac{\partial}{\partial x} F(x) \right) + \frac{4\alpha}{\Delta s^4} \mathrm{I}_3 \right]^{-1} \quad (3.8)$$

$$h_i^{k+1} = \left[\sum_{x \in R} \left(\frac{\partial}{\partial x} F(x') \right)^T W(x) (G(x) - F(x')) + \frac{2\alpha}{\Delta s^4} \left(h_{i-1}^k + h_{i+1}^k \right) \right] \times \left[\sum_{x \in R} \left(\frac{\partial}{\partial x} F(x') \right)^T W(x) \left(\frac{\partial}{\partial x} F(x') \right) + \frac{4\alpha}{\Delta s^4} \mathrm{I}_3 \right]^{-1} \quad (3.9)$$

The solution to h_i is dependent on h_{i-1} and h_{i+1}, and so we solve for h_i by iterating (3.9), where $h_i^0 = 0$ and $x' = x + h_i^k$. However, as opposed to using a single window size to define R, we first use a large window to account for large displacements that may occur from frame to frame. We then use progressively smaller window sizes to stabilize and converge to a more accurate solution for h_i. For this study, we execute 50 iterations at window sizes of 11^3, 9^3, 7^3, 5^3, and 3^3, in that order. The tracked annulus, which is tracked from frame to frame starting from a closed frame, is then interpolated (spline interpolation) to create a continuous contour in space.

3.2 Results

The 4DMAS method consists of three major components: a valve state predictor, an annulus segmentation algorithm (3DMAS method), and an annulus tracking algorithm (CLKOF method). As we presented validation of the 3DMAS algorithm in Chapter 2, we

Table 3.1: Summary of clinical mitral valve images used in the validation studies.

Valve	Mitral Valve State	Frames in Sequence
1	Prolapse; Regurgitation	12
2	Prolapse; Dilated Annulus	20
3	Normal	11
4	Normal	15
5	Cleft Mitral	16
6	Normal	13
7	Prolapse	13

present here an analysis of the valve state predictor and annulus tracking method. In the analysis, we used seven anonymized clinical transesophageal echocardiography sequences (iE33 Echocardiography System with X7-2t transesophageal probe, Philips Healthcare, Andover, MA, USA). To capture the entire mitral valve throughout the ultrasound sequence, these had to be acquired as EKG-gated full volume reconstructions. Volume dimensions were roughly $200 \times 200 \times 200$ voxels with a resolution of 0.5–0.75 mm/voxel. More details about the images and valves can be seen in Table 3.1.

An example of the results of the 4DMAS method are shown in Figure 3.4 for the example sequence shown in Figure 3.2. As a comparison, also shown are the results when tracking is performed with the original LKOF method versus the CLKOF method, both using the window sizes and iterations as described. This highlights the stability of the CLKOF method versus the LKOF method for annulus tracking in ultrasound.

The algorithm was coded mostly in MATLAB (The Math-Works, Natick, MA, USA), but used C++ for both computing parts of the thin tissue detector and for the implementation of the max-flow algorithm. In the validation studies, the parameter values used in the 3DMAS algorithm are the same as those listed in Chapter 2. The approximate time to compute the annulus for closed and open valves (64-bit PC, 3.0 GHz Intel Core 2 Duo processor with 4GB of RAM) was roughly 90 and 30 seconds, respectively, which includes the time to determine valve states.

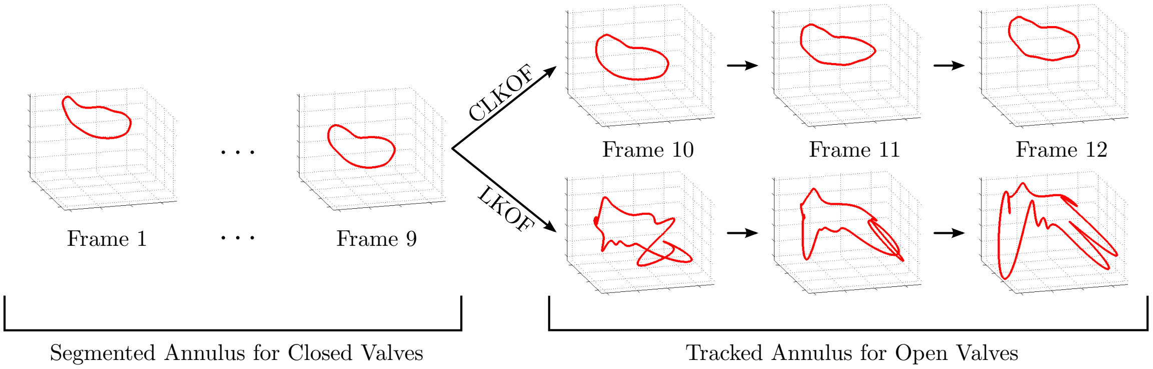

Frame 1
· · ·
Frame 9
CLKOF
LKOF
Frame 10
Frame 11
Frame 12
Segmented Annulus for Closed Valves
Tracked Annulus for Open Valves

Figure 3.4: Comparison of the resulting 4D annulus delineation for the example 4DUS sequence shown in Figure 3.2 when tracking is performed using the original LKOF method (i.e. minimizing (3.2)) versus the CLKOF method (i.e. minimizing (3.3)).

3.2.1 Valve State Predictor Accuracy Study

Essential steps in computing the valve state predictor are finding the mitral leaflet surface in every frame, and then cropping the surfaces by the projection of A_{ref} segmented from the user-specified reference frame, F_{ref}. The valve state predictor is then dependent on F_{ref}. The user-specified point has no effect on the predictor, as the annulus segmentation in a closed frame is operator-independent for any reasonably placed point. We therefore present here an analysis of the valve state predictor's accuracy and how it is affected by the user's choice of the reference frame.

For this study, we manually selected those frames that contained a closed valve (closed frames) versus those that contained an open valve (open frames), where a closed valve was defined to be a valve with leaflets completely coapted. We considered each closed frame as a potential candidate for F_{ref} and subsequently computed the valve state predictor as described in Section 3.1.2 using each of these frames as F_{ref}. Because computing the valve state predictor is dependent on automatically computing A_{ref} using the 3DMAS method, which can only be done for closed valves, we did not consider open frames as a potential candidate for F_{ref}. The collective accuracy of the valve state predictor across

Table 3.2: Valve state predictor accuracy across all potential reference frames. *Closed Valves* and *Open Valves* are those frames manually selected to contain a closed or open mitral valve, respectively. %Closed and %Open indicate the percentage of frames in a category that the valve state predictor labeled as containing either a closed or open valve, respectively.

Data	*Closed Valves*		*Open Valves*	
Set	%Closed	%Open	%Open	%Closed
1	100	0	100	0
2	100	0	100	0
3	100	0	100	0
4	93.1	6.9	100	0
5	100	0	100	0
6	100	0	100	0
7	100	0	100	0
Group	99.2	0.8	100	0

all potential reference frames is shown in Table 3.2. This table shows that the valve state predictor was on average 99.2% accurate in identifying closed frames and 100% accurate in identifying open frames, suggesting that the predictor is insensitive to the choice of F_{ref} when it contains a closed valve.

3.2.2 3D Annulus Tracking Validation

CLKOF Method vs. Human Observers

To assess the accuracy of the tracking method, we compared the annulus tracking results to manual delineations made by a group of three experts. We provided the experts with slices from 4DUS of the mitral valve, where slices were taken at 10 degree increments about the mitral axis. Within each image the experts had access to temporal information available from the entire ultrasound sequence.

As the 4DMAS method operates by tracking to open frames the annulus segmented automatically from a closed frame, we similarly asked the experts to do the same. We did this by providing the experts with the location of the automatically segmented annulus in the closed frames immediately neighboring the open frames. We then asked the experts to

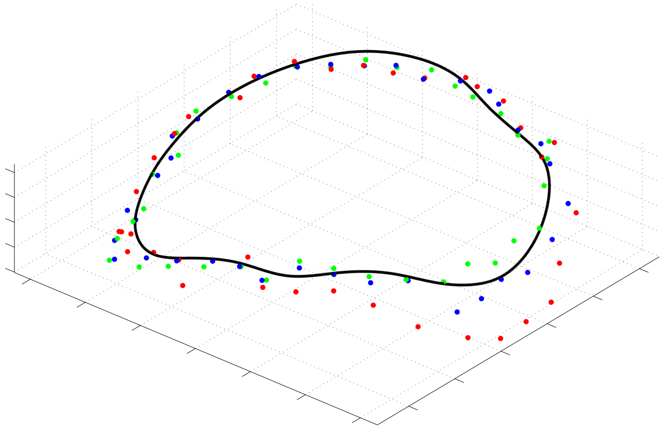


Figure 3.5: Typical comparison between the annulus tracked by the human observers (colored points) and that tracked using the CLKOF method (solid contour).

track the location of the displayed annulus to the neighboring frames containing an open mitral valve by clicking on the tracked annulus point locations in each slice. From the seven data sets, 30 frames total were used for this study. The manual tracking of an annulus from one frame to a neighboring frame took roughly 5 minutes on average across all participants and all tracked frames. A typical comparison between a manually and automatically tracked annulus is shown in Figure 3.5.

We computed accuracy measures for the CLKOF method by measuring the distance of the automatically tracked annulus to the mean of the points delineated by the experts. The results from the analysis for each data set can be seen in Table 3.3, where it is shown that across the seven data sets (30 frames total), there was an average RMS difference of 1.67±0.63mm of the algorithm annulus to the mean of the experts.

The previous analysis treats the expert mean as a gold standard as a means of computing an error for the CLKOF method, however, it is also important to gauge the relative performance of the group of experts and the algorithm to determine if there is a discernible difference between the two groups. To perform this analysis, we repeated the analysis that was done for the CLKOF method for each expert. The difference being that

Table 3.3: RMS of the Euclidean distance of the automatically tracked annulus from the mean of the manually tracked annulus points.

Data Set	RMS Distance (mm) (mean ± std. dev.)	No. of Open Frames
1	1.96 ± 0.70	3
2	0.75 ± 0.06	4
3	2.30 ± 0.40	5
4	1.04 ± 0.07	5
5	1.71 ± 0.26	4
6	2.02 ± 0.38	6
7	1.86 ± 0.44	3
Group	1.67 ± 0.63	30

Table 3.4: RMS of the Euclidean distance of each experts' manually tracked annulus from the mean computed from the other two experts' manual segmentations and the CLKOF tracked annulus.

	RMS Distance (mm)			No. of
Data	(mean ± standard deviation)			Open
Set	Expert$_1$	Expert$_2$	Expert$_3$	Frames
1	2.04±0.30	3.02±0.07	2.40±0.51	3
2	0.83±0.13	1.81±0.08	1.16±0.05	4
3	2.05±0.19	3.01±0.22	5.62±0.86	5
4	0.66±0.06	1.20±0.17	1.14±0.06	5
5	1.02±0.21	1.98±0.08	1.69±0.34	4
6	1.83±0.41	2.60±0.36	1.50±0.24	6
7	1.49±0.33	3.01±0.62	1.69±0.05	3
Group	1.99 ± 1.15			30

in this study, a given expert's segmentation was compared to the mean computed using the segmentations from the other two experts and the CLKOF algorithm, thereby treating the algorithm as an equal participant. The results of this analysis can be seen in Table 3.4. While the overall performance of the CLKOF method (RMS = 1.67±0.63mm, $n = 30$) was slightly better than that of the group of experts (RMS = 1.99±1.15mm, $n = 90$) in that the CLKOF method had a lower mean RMS distance, the difference between the two groups is not statistically significant ($p = 0.062$), suggesting that the CLKOF method performed at least as well as the group of experts.

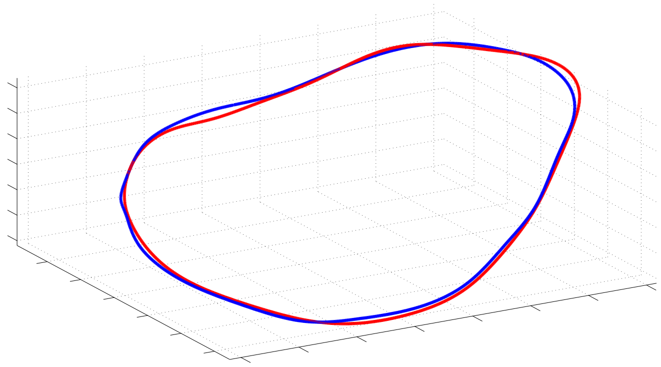


Figure 3.6: Typical comparison between an annulus delineated using the CLKOF tracking method (blue) and the 3DMAS method (red) for a closed valve.

CLKOF Tracking Method vs. 3DMAS Method for a Closed Valve Annulus

We showed in Section 3.2.1 that when the valve state predictor mislabels a valve, it fails by mislabeling a closed valve as an open valve (Table 3.2, Data Set 4). In this case, the CLKOF method is employed instead of using the 3DMAS algorithm. As the 3DMAS algorithm was shown in Chapter 2 to be highly accurate, it is important to know how much an automatically segmented annulus and a tracked annulus differ for a closed valve. A typical comparison of the resulting annuli from the two methods is shown in Figure 3.6.

In this study, for each data set we randomly chose a frame with a closed valve and tracked the annulus to neighboring frames with a closed valve using the CLKOF method. Additionally, we computed the automated annulus segmentations for the same neighboring frames using the 3DMAS method. We then compared the different annulus delineations by computing the RMS distance between the two annulus point sets. The results of the analysis are shown in Table 3.5, where the average RMS difference across all data sets was found to be 0.58±0.31mm, which is on the order of the volume resolution.

Table 3.5: Distance between an annulus delineated using the CLKOF tracking method from an annulus segmented using the 3DMAS method for closed frames $n-1$ and $n+1$ neighboring the randomly selected frame n.

	RMS Distance (mm)	
Data Set	Frame$_{n-1}$	Frame$_{n+1}$
1	0.38	0.45
2	0.31	0.48
3	0.55	0.68
4	0.34	0.50
5	0.78	0.59
6	0.38	0.56
7	0.48	1.56
Avg.±Std. Dev.	0.58 ± 0.31	

3.3 Discussion

3.3.1 Performance and Validation

The 4DMAS method is unique in that it constructs a 4D annulus by first detecting whether a valve is open or closed in a given frame so that the segmentation approach can be appropriately altered. We showed that the valve state predictor is highly accurate, with over 99% correct assignment to the correct state in validation tests. When the valve state predictor did not coincide with the manual clustering, it did so by predicting that a closed valve was actually an open valve, and was the result of the valve being in a transition state. In this case, the CLKOF method would be used to track the annulus from a neighboring frame to the mislabeled frame instead of using the 3DMAS method to automatically segment the annulus. We showed in our analysis that the average RMS difference between a tracked (CLKOF) annulus to a segmented (3DMAS) annulus for closed valves was 0.58±0.31mm, indicating that the penalty associated with mislabeling a closed valve as an open valve is small. A more significant failure would be if the valve state predictor mislabeled an open valve as a closed valve. In this case, the 3DMAS algorithm would be employed for an open valve, and because the 3DMAS algorithm was not designed for open valves, the error associated with this mislabeling is substantial.

The accuracy of the 4DMAS method is not only a result of being able to predict the valve state, but having predicted the valve state, using accurate methods to find the annulus. We showed in Chapter 2 that the 3DMAS algorithm is accurate and robust when finding the annulus for closed mitral valves. In this chapter, we showed that the CLKOF tracking method is as accurate at segmenting the annulus of open valves. In comparing the automated tracking results to manual delineations made by a group of experts across 30 frames, we found an average RMS difference of 1.67±0.63mm. This is on the same order of accuracy as the 3DMAS method, which had an RMS difference of 1.81 ± 0.78mm when compared to manual annulus delineations made by a group of 10 experts on 10 different valves. Additionally, we showed that when comparing the overall performance of the CLKOF algorithm (RMS = 1.67±0.63mm, $n = 30$) to that of the group of three experts (RMS = 1.99±1.15mm, $n = 90$), while the algorithm tended toward better performance, there was no statistically significant difference between the two groups ($p = 0.062$).

The CLKOF tracking method performs well for annulus tracking in ultrasound because it can compensate for noise and inhomogeneities in the ultrasound volume by taking advantage of the correlation between the motion of neighboring points. Referring back to (3.8), which is the solution for the displacement, h_i, of point i along the annulus, it follows that the contribution of the imaging and the prior knowledge about annular geometry are controlled by the scalar weight α. In setting α to zero, the solution for h_i is the same as originally derived by Lucas & Kanade [57]. In this case, tracking results are dependent entirely on the imaging. An example of the resulting annulus can be seen in the bottom row of Figure 3.4. Conversely, if α is made to be large or if the imaging provides little information (i.e. $\frac{\partial F}{\partial x} \approx 0$), then $h_i \approx \frac{1}{2}\left(h_{i-1} + h_{i+1}\right)$ (i.e. the displacement is the average of the displacement of the neighboring points). Therefore, assuming α is neither too small nor too large, the annulus tracking would be dictated mostly by the imaging in areas of high definition in the ultrasound volume, and by the geometric prior in areas of low definition. The result is a more stable and accurate tracking of the annulus. The implementation of a constrained optical flow algorithm has also shown to be useful in other previous applications such as face tracking [15] and left ventricle motion detection in MR images [53].

When compared to previous automated methods for 4D annulus segmentation from ultrasound ([98] and [37]), the presented method offers notable improvements in several areas, the first of which is the amount of overhead and degree of user-interaction. Previous methods either required manual initialization of an annulus, intermittent manual corrections of an automated algorithm, or a large database of manually selected feature locations. Conversely, the 4DMAS method requires only the selection of a frame with a closed valve and a point near the center of that valve. Secondly, because of the accuracy of the valve state predictor and the insensitivity of the 3DMAS method, the results of the 4DMAS method are likewise insensitive to the user's input, resulting in an operator-independent 4D annulus segmentation. Lastly, the extensive analysis performed both on the 3DMAS method, on the valve state predictor, and on the CLKOF tracking method make the 4DMAS method the most well validated 4D annulus segmentation algorithm.

3.3.2 Algorithm Design

As alluded to in Section 3.1.1, the 3DMAS method is designed for ultrasound images where the thin leaflet tissue is visible. Additionally, both the 3DMAS and CLKOF methods are designed for when the entire mitral annulus is contained in the conical ultrasound volume. As a result, the 4DMAS method is designed for ultrasound images acquired with an *en face* view of the mitral valve and that contain the complete mitral annulus in the conical data volume throughout the cardiac cycle. This is why the 4DMAS validation studies used EKG-gated full volume reconstruction images of the mitral valve. However, if similar views could be obtained with the higher temporal resolution as seen in live 3D acquisitions, these would be preferred.

While the 4DMAS relies on imaging to predict valve state, one might argue that using an EKG signal might prove to be simpler and just as effective, as there should be a correspondence between the signal and the valve state. However, there are many reasons why an image-based valve state predictor is preferred over the EKG signal. One reason is that the EKG signal can be adversely affected by many variables in the operating-room environment. Additionally, depending on disease states, the electrophysiology of the heart may not correspond to the valve state. For instance, in the case of systolic anterior motion, the mitral valve may in fact be pulled open during systole. It is then more reliable to use the appearance of the heart in imaging to predict state rather than make predictions based on correlations to electrophysiology.

In the CLKOF method, it is worth noting that the scalar weight α was tuned to a constant of 500, as this was found to adequately balance both the contribution of the imaging and the geometric prior. However, α need not be set to a constant, but could be determined dynamically as a function of position of an annulus point along the contour, local curvature of the contour, or local imaging characteristics. Given the accuracy of the current method, however, the benefit of this dynamic assignment is likely small.

A current limitation of the 4DMAS method is that, due to separate executions of the 3DMAS method on closed frames, a correspondence between annular points of neighboring closed frames is not inherent. Therefore, while the annular geometry is tracked as a whole throughout the cardiac cycle, specific annular points are not. We could compensate for this, however, by using geometric constraints and similarity measures, such as ultrasound intensity or thin tissue detector patterns, to determine a correspondence between annuli after they have been segmented.

3.3.3 Future Work

The mitral annulus is an important cardiac structure that is widely studied both clinically and for research purposes. The presented method, being that it is accurate and requires little user interaction, could serve to expedite a number of new studies. Knowing the valve state in a frame could also prove valuable in future work to determine the temporal location of a frame relative to the cardiac cycle without using an electrocardiographic (ECG) signal. This could be useful information in related segmentation efforts, such as mitral leaflet or left ventricle segmentation. Also, because the presented method can determine valve state and switch between direct segmentation and tracking, it does not accumulate substantial tracking error, and could be used to analyze longer ultrasound sequences than those studied.

Future work will also include addressing the limitations on the images for which the 4DMAS method will successfully and accurately segment the mitral annulus - in particular, the limitation that the entire mitral annulus needs to be contained in the ultrasound volume throughout the cardiac cycle. Currently, to obtain such an ultrasound sequence, ECG-gated full volume reconstructions are acquired. However, this consequently lowers the frame rate. We show in later chapters how to overcome these issues and construct large field-of-view ultrasound images with high temporal resolution.

Chapter 4

Three-Dimensional Mitral Leaflet Segmentation for Open Valves

The methods for modeling the leaflets from 3DUS can be categorized as either volumetric, mesh, or parametric. An example of a volumetric method for 3DUS was shown in [85], which used an intensity-based level set method to locate the leaflets. Unfortunately the method included not only the leaflets but anything connected to the leaflets with similar intensity, such as the left ventricle wall, thus failing to isolate the leaflets. A method for modeling the valve as a mesh was presented in [10], which used a thin tissue detector and level sets to locate the leaflets and surrounding tissue. The method subsequently required manual intervention and the assumption of a planar annulus to construct an isolated leaflet mesh geometry. The method presented in [37] segmented and tracked the locations of several features of the mitral and aortic valve structures in 3DUS using machine learning techniques. It then fit a parametric model to the segmented locations. However, in fitting the model to only a few points, the model lacked patient-specific detail.

To compensate for the highlighted insufficiencies of previous methods, we have developed an automatic method for modeling the leaflets. The method is designed for open mitral valves because we want to differentiate between anterior and posterior leaflets, and making this distinction for coapted leaflets in 3DUS is extremely difficult even for a human observer. After finding the location of the annulus using the method outlined in Chapter 3, the leaflet modeling method automatically finds the location and extent of the mitral leaflets and generates a mesh to represent the geometry with no additional user interaction. We chose to represent the leaflets as a mesh because this representation can be easily used to generate either a volumetric or parametric representation if needed, is a suitable input for mechanical modeling, and is representative of the the thin leaflet tissue.

The method uses the location of the annulus as a means to enforce knowledge about the location and orientation of the leaflets. The method then uses a series of steps to estimate and refine a search space coordinate system and the shape and location of the leaflets. A mesh at the location of the leaflets is then constructed as a natural consequence of the geometry of the search space. The details of the algorithm are described in Section 4.1. A study quantifying the performance of the method as compared to manual leaflet tracings is then shown in Section 4.2.

4.1 Methods and Materials

4.1.1 Algorithm Summary and Components

The mitral leaflet modeling algorithm is comprised of five main components organized in a two-pass strategy that drives the algorithm to accurately locate the leaflets (Figure 4.1). The components can be described as defining the search space, redefining the search space axis based on the availability of new information, estimating an extended leaflet surface using a graph cut method, refining the extended leaflet surface using an ac-

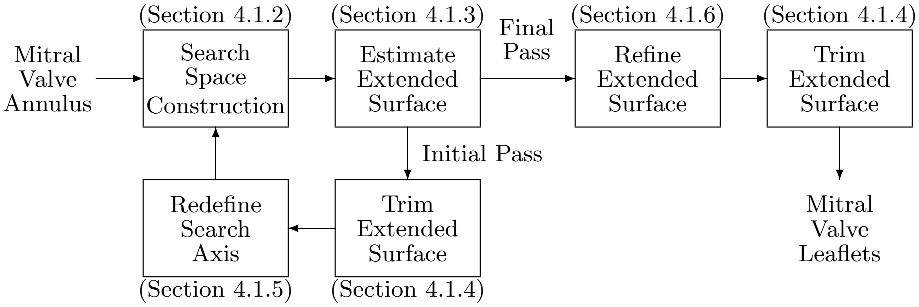


Figure 4.1: Flow chart summary of the leaflet modeling algorithm. A description of each component appears in the corresponding section.

tive surface method, and finally trimming the extended leaflet surface using dimensional reduction and a level set method.

4.1.2 Constructing the Search Space

The location of the mitral leaflets in the heart is such that they attach to and rotate about the mitral annulus. Therefore, we search for the location of the mitral leaflets by constructing a search space that is comprised of an arc system that is centered about the annulus (Figure 4.2). The arcs are constructed in planes that are rotated about a defined axis. Initially, little is known about the leaflets aside from their orientation, and so the axis is generically defined as the direction of least variance as determined from a principal component analysis of the mitral annulus. For the final pass, the axis location is refined, as described in Section 4.1.5, to account for the estimated leaflet orientation and position. The search planes are evenly spaced at an angular offset of $\Delta\Theta_p$. The arcs are evenly spaced at a radial offset of ΔR, and points are sampled along the arcs at an angular offset of $\Delta\Theta_s$. The search space is designed around the assumption that the leaflets will intersect the search arcs, at most, at a single location along each arc. This point along each arc is found first using a graph-cut method to estimate the leaflet location, and in the final pass, the location is refined to account for surface curvature and the image.

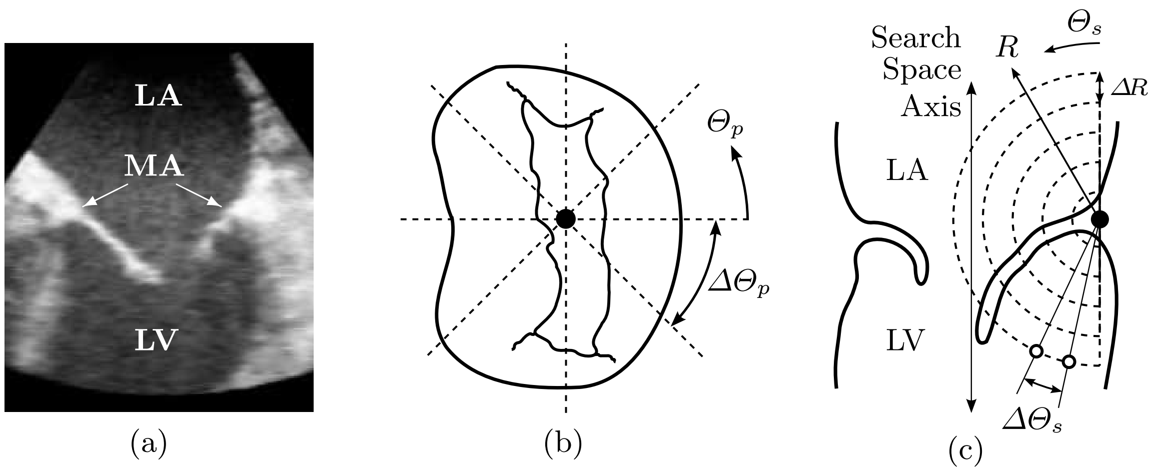


Figure 4.2: Example of the location (relative to the mitral valve) and spacing of the search space axis, planes, arcs, and sample points. (a) Open mitral valve in a clinical 3DUS slice. (b) Atrial view of an open mitral valve showing search planes oriented about the search axis. (c) Cut-plane view of an open mitral valve showing the search arc and sample point configurations. (LA - left atrium; LV - left ventricle; MA - mitral annulus)

4.1.3 Estimating an Extended Leaflet Surface

The location of the leaflets is estimated by fitting a surface to their believed location. The surface is found as the min-cut of a graph [7], where the source connects to all points at the minimum Θ_s and the sink to all points at the maximum. The undirected edges of the graph, Γ, connect neighboring points at $\Theta_p \pm \Delta\Theta_p$, $R \pm \Delta R$, and $\Theta_s \pm \Delta\Theta_s$, making Γ 6-connected. The edges of the graph have a capacity that is a function of a drive image, which is computed at each point. The drive image on the first pass is a product of the original 3DUS intensity ($\mathbf{I}$) and a thin tissue detector ($\mathbf{TTD}$) [83], $\mathbf{I}_{drv} = (\mathbf{I})(\mathbf{TTD})$. For the final pass, because we have an estimate for the leaflet location, a weighting function, $\mathbf{W}$, is incorporated into the drive image, $\mathbf{I}_{drv} = (\mathbf{I})(\mathbf{TTD})(\mathbf{W})$, to encourage the min-cut to be found at the estimated leaflet location. The edge capacity between points i and j is then defined as

$$E_{ij} = \frac{1}{1 + \omega\left(\mathbf{I}_{drv,i} + \mathbf{I}_{drv,j}\right)^2}, \tag{4.1}$$

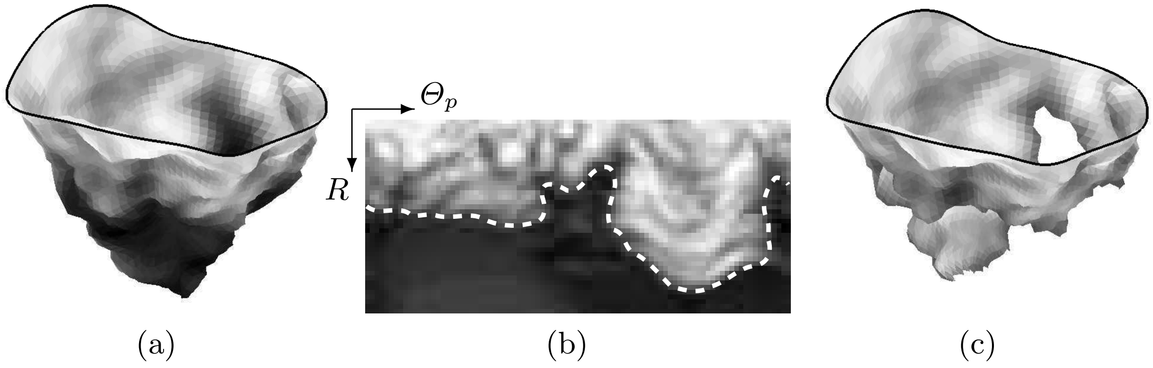


Figure 4.3: (a) Example of an extended and refined leaflet surface. The gray level of the triangles reflect the 3DUS intensity. (b) The intensity image formed from the ultrasound intensity at the surface nodes. Also shown is the computed trimming contour (dashed white line) found using a level set formulation. (c) The surface from (a) that is trimmed according to the contour shown in (b). The black contours in (a) and (c) indicate the location of the mitral annulus.

where ω is a scalar weight. The min-cut of the graph is found using Kolmogorov's implementation [45]. The point representing the leaflet surface location along each arc is herein referred to as a leaflet node.

4.1.4 Trimming an Extended Leaflet Surface

The extended leaflet surface should reside at the location of the leaflets and extend past the edges of the leaflets into the blood pool of the left ventricle. It is therefore necessary to trim the surface at the leaflet edge so that the remaining leaflet geometry accurately portrays the location and geometry of the mitral leaflets. The trimming operation is done by isolating the nodes that reside at the leaflets from those that reside in the blood pool. This is done by first reducing the ultrasound intensity information at the surface nodes to a two-dimensional image, where the axes of the image are the search plane angle and the arc radial offset (Figure 4.3). The leaflet nodes are then separated from the blood pool nodes using the RCAC level set method [85], where the level set function is initialized using a k-means algorithm. The annulus nodes are always made to be included in the leaflet group during the level set evolution, and the continuity between the first and last search plane is maintained. Nodes that are not found to be a member of the leaflet group are considered to be part of the blood pool and removed from the surface.

4.1.5 Redefining the Search Space Axis

The search space axis is initially defined with only the knowledge about the general orientation of the leaflets. The problem with this generic assignment is that information about the leaflet position is unknown, and so the axis could potentially pass through the leaflet tissue, thereby creating an inconsistency in the estimated leaflet location. However, as information about the estimated leaflet location becomes available, the axis can be better defined to account for the position of the leaflets. This is done by making the axis the vector that passes through the mitral annulus center point and the center of the estimated leaflet opening. In doing so, the axis will not pass through any leaflet tissue, and will provide for a more appropriate search space in which to find the leaflets.

4.1.6 Refining an Extended Leaflet Surface

The min-cut algorithm is well suited for estimating the location of the extended leaflet surface, but oftentimes fails to capture finer leaflet detail. Therefore, a surface refinement method treats the surface as an active surface and drives the it toward the location of the leaflets in the image and also regulates surface curvature. In the refinement process, the nodes are restricted to move along their respective arcs in the search space. An image energy, $\mathbf{E}_{img}$, which is the inverse of the ultrasound intensity, drives surface nodes to the leaflet location along the arc. An internal energy, $\mathbf{E}_{crv}$, regulates surface curvature by driving surface nodes to locations that will result in uniform edge length. The minimized surface energy equation is then $\mathbf{E}_{surf} = \omega_{img}\mathbf{E}_{img} + \omega_{crv}\mathbf{E}_{crv}$, where ω_{img} and ω_{crv} are scalar weights.

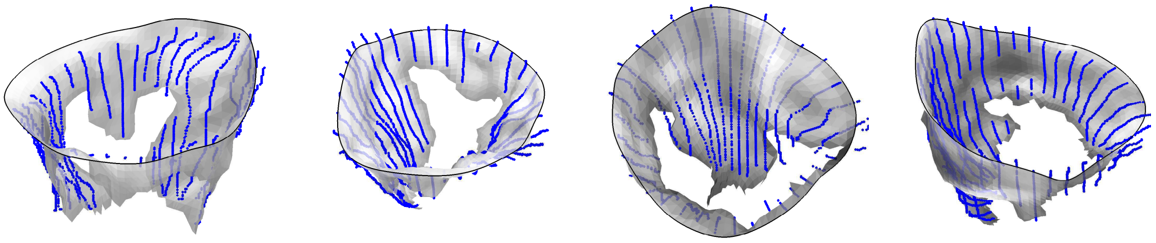
4.1.7 Mesh Generation

The nodes of the trimmed leaflet surface are formed into a triangular mesh by first connecting nodes to those neighbors that exist at $\Theta_p \pm \Delta\Theta_p$ and $R \pm \Delta R$. Diagonal edges are then defined between nodes at (Θ_p, R) and $(\Theta_p + \Delta\Theta_p, R + \Delta R)$. The normals of the triangles are defined such that the positive normal direction points toward the top (atrial) side of the mitral leaflets.

4.2 Performance and Validation

We assessed the performance of the mitral leaflet modeling algorithm on a group of four anonymized clinical pediatric cases. The mitral valves in these cases were imaged using transesophageal echocardiography (iE33 Echocardiography System with X7-2t transesophageal probe, Philips Healthcare, Andover, MA, USA). A frame showing an open valve was manually selected from each of the four cases. The annulus, which acted as the input to the algorithm, was found using the method described in Chapter 3. Using the location of the tracked annulus in the manually selected frame with an open valve, the leaflets were then found using the presented method.

The leaflets in the chosen frames were manually traced in cut planes taken every 10^o about the search space axis and compared to the meshes generated by the algorithm (Figure 4.4). The intersection of the leaflet mesh with the cut planes generated contour segments in the plane. A histogram of the distances of points along these contours from the manually traced leaflet contours were then computed (Figure 4.5). As indicated in the histogram plot, on average 95% of the leaflet mesh points resided within 2.10mm of the manually traced leaflets, with the average distance being roughly 0.76 ± 0.65mm. These



(a) Valve 1 (b) Valve 2 (c) Valve 3 (d) Valve 4

Figure 4.4: Comparison of the algorithm-generated leaflet model (shaded surfaces) to manual tracings (contours) made in cut planes about the search space axis.

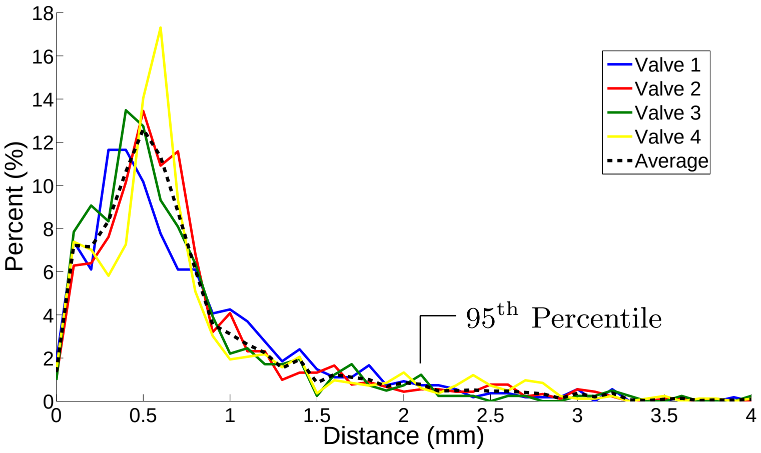


Figure 4.5: Histogram of the distances of the algorithm mesh from the manually traced points in each cut plane for each valve in Figure 4.4 and for the group average. The 95th percentile is shown for the group average.

errors were on the order of the ultrasound volume resolutions (0.5-1.0mm/voxel) and leaflet thicknesses observed in the volumes (2-3mm).

4.3 Discussion

This chapter presents a method that automatically generates a model of the leaflets for an open mitral valve in a 3DUS volume given the location of the mitral annulus. The presented method differs from previous leaflet modeling methods in that it does not require extensive user interaction and generates a detailed, patient-specific leaflet mesh. The method uses a two-pass strategy that defines and refines a search space coordinate system and the leaflet surface to generate a mesh at the location of the leaflets. We showed the accuracy of this method by comparing the mesh to manual tracings of the leaflets made in cut planes taken about a designated axis, where it was found that 95% of the points along the mesh were less than 2.10mm away from the manual tracings.

The leaflet modeling is robust and accurate partly due to the fact that it exploits prior knowledge about where to find the leaflets based on the location of the mitral annulus. Accurately locating the annulus is therefore important as its location directly affects the leaflet model. To find the annulus, we used the annulus segmentation and tracking method described in Chapter 3. This method has a segmentation error of 1.81±0.78mm and a tracking error of 1.67±0.63mm when compared to manual delineations made by a group of experts. The accuracy of both the annulus and leaflet modeling methods makes for one of the most accurate and robust mitral valve modeling methods capable of capturing detailed patient-specific valve geometry.

A limitation of the presented leaflet modeling method is that the 3DUS images of the mitral valve need to show the leaflets as a thin structure in an open valve state. This suggests that short-axis views of the valve will not suffice, as the chordal structure that becomes visible in this approach gives the leaflets a thick appearance. Also, the valve cannot be completely open (peak diastole) as the leaflets are pushed up against the left ventricle wall and are typically indistinguishable from the wall as seen in 3DUS. The recommended images are then those that use either an *en face* or apex view of the mitral valve and capture the valve during the process of opening or closing. In these views, the anterior and posterior leaflets should be differentiable.

In a later chapter we describe how to use the defined leaflet mesh as a geometric prior to be tracked to other frames in a 3DUS sequence to provide for a four-dimensional image-driven model of the leaflets. The benefit in this approach is that anterior and posterior mitral leaflets are differentiated throughout a sequence, and because leaflet geometry are preserved and collisions resolved, a coaptation line and surface are found. However, for accurate leaflet tracking, the progression of valve closure needs to be able to be observed in a 3DUS sequence. As native 3DUS acquisitions do not have sufficient temporal resolution to make this observation, we explain in the next chapter how to first create high temporal resolution 3DUS sequences.

Chapter 5

Reconstructing High Temporal Resolution 3D Echocardiography

To be able to track the leaflets so that the anterior and posterior leaflets can be differentiated throughout a cardiac cycle and a detailed coaptation region delineated, we need to be able to observe the progression of valve closure. As native 3DUS sequences do not have sufficient temporal resolution to allow this observation, we need to improve the temporal resolution of three-dimensional ultrasound (3DUS) imaging.

When using 3DUS to image the structures of the heart, there are several trade-offs between the spatial resolution (SR), temporal resolution (TR), and field of view (FOV) which have to be negotiated. These trade-offs are a consequence of how an ultrasound volume is formed, which is as a compilation of scan lines in a given region of interest. Because each scan line takes a specified amount of time to acquire, it follows that the more scan lines that are to be acquired (which may serve to either increase the FOV or increase the scan line density, thereby improving SR), the longer it will take to acquire a volume. The result is a lower temporal resolution. Conversely, a smaller FOV or lower SR will serve to improve the TR.

A problem with these trade-offs arises when studying heart valve function, and in particular, their opening and closing. The valves in the heart open and close in tens of milliseconds [96]. In contrast, the time to acquire a 3DUS volume of an entire valve with adequate spatial resolution can take just as long, if not longer. Volume acquisition times for transesophageal 3DUS can take anywhere from tens to hundreds of milliseconds, depending on volume size. As such, accurately capturing intermediate stages of the opening and closing of valves is difficult.

To better image valve motion, several studies have exploited the periodicity of heart motion and the correlation between the time the valve opens and closes with the electrocardiography (ECG) signal. The most prevalent of these studies are those that use prospective ECG-gating [62, 69, 110]. These studies use the real-time ECG signal to trigger when 3DUS volumes are acquired relative to the cardiac cycle, allowing, over several cardiac cycles, for multiple frames to be acquired during valve opening and closing (Figure 5.1). Image-based methods that generate a pseudo-ECG signal have also been used [42, 9]. A disadvantage of prospective gating is that it can take a long time, oftentimes several minutes, to acquire the necessary data to accurately image a valve over a full cardiac cycle. Another approach is retrospective ECG-gating (Figure 5.1), which as opposed to controlling the time of data acquisition relative to the cardiac cycle as done in prospective ECG-gating, collects data at a fixed rate and and then reorders the data according to its acquisition time relative to the cardiac cycle using the QRS complex as seen in the ECG signal [50, 11, 16, 39, 90]. The assumption of this method is that if volumes are acquired over enough cardiac cycles, adequate coverage over the cardiac cycle will result and the time between frames will be a few milliseconds. However, unlike prospective ECG-gating methods, a uniform and complete sampling across the cardiac cycle cannot be guaranteed.

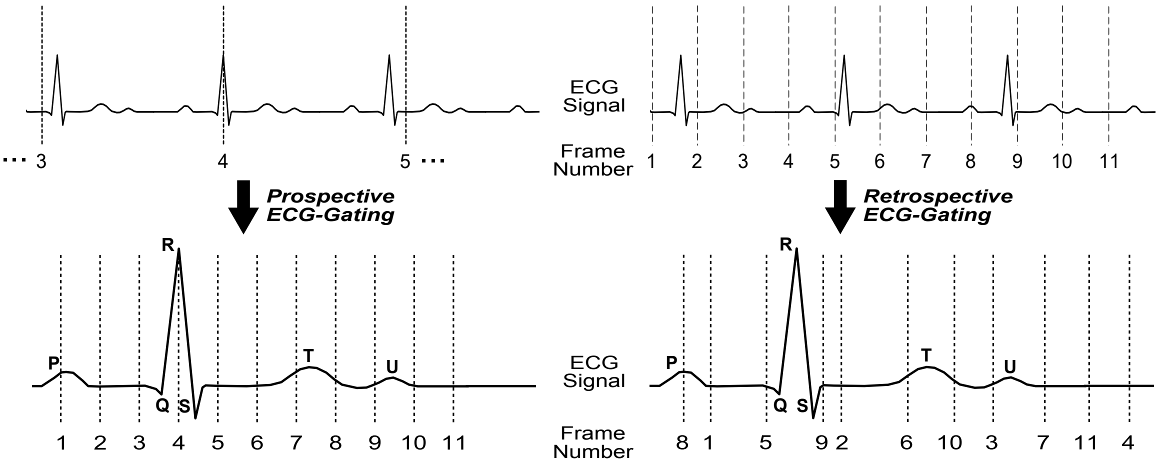


Figure 5.1: Simplified summary of prospective and retrospective ECG-gating methods for ultrasound. (*Left*) Prospective ECG-gating acquires frames at known times (start of frames shown by dashed lines) during the cardiac cycle using the ECG signal to ensure adequate and even sampling over a single cardiac cycle. (*Right*) Retrospective ECG-gating acquires frames at a fixed rate and then reorders the frames according to their start times relative to the cardiac cycle using the ECG signal. Retrospective gating does not guarantee adequate and even sampling across a cardiac cycle, but sequence acquisition times are generally shorter than prospective gating because data is continuously acquired.

Methods like prospective and retrospective ECG-gating are able to acquire or place a frame in the correct temporal location in a cardiac cycle according to the start time of each frame. In doing so, the start time between frames (sequence temporal resolution, STR) can be made small, oftentimes on the order of milliseconds. However, the time to acquire the volumetric data in each frame (frame time interval, FTI) is tens to hundreds of milliseconds. This disparity between the STR and FTI results in a temporally incoherent sequence in that the end time of a frame occurs after the start time of the next resequenced frame (Figure 5.2). Another problem is that both the prospective and retrospective ECG-gating methods require acquiring data over several seconds or minutes, but in that time the heart is likely to shift relative to the ultrasound probe, either due to respiration or other external forces. Lastly, in the case of prospective ECG-gating, the acquisition time alone renders the method impractical for clinical use.

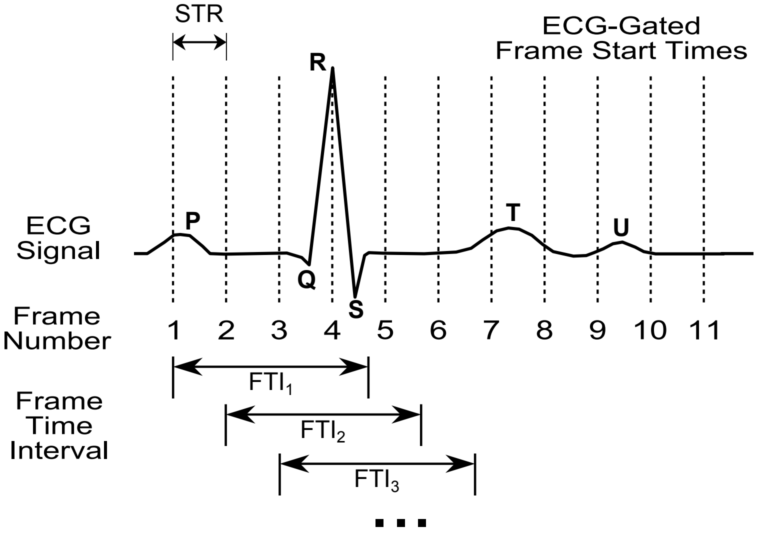


Figure 5.2: Simplification of the temporal incoherence problem in ECG-gated sequences. The time between the start of gated frames (i.e. sequence temporal resolution, STR) is small, but because the frame acquisition time (i.e. frame time interval, FTI) is large, the end of a gated frame can occur after the beginning of the next gated frame. While the example uses the prospective ECG-gated sequence from Figure 5.1, the same problem would exist for the retrospective ECG-gated sequence.

The presented reconstruction method generates high temporal resolution 3DUS sequences from standard acquisitions and accounts for the drawbacks of previous methods. The method uses retrospective ECG-gating, and also incorporates image stabilization and temporal distortion correction (TDC) algorithms. Using retrospective gating, frames are placed in order according to their start time relative to the cardiac cycle. The frames are then spatially shifted to account for small shifts that may have occurred during the original acquisition due to respiration or other external factors. A temporal distortion correction (TDC) algorithm then uses the resequenced and stabilized images, as well as an assumption about the temporal profile of each frame, to recompute the values at each voxel in each frame so that the STR and FTI of the resulting sequence more closely align. Benefits of the presented reconstruction method are that it can be performed on data acquired by most ultrasound machines with no additional equipment, it takes little time to acquire the necessary data, patients do not need to be put on breath hold (i.e. small shifts frame-to-frame can be accounted for), and the resulting temporal resolution of the reconstructed sequence can be on the order of milliseconds, depending on the length of the original acquisition. Additionally, when using a framework that incorporates a graphics processing unit (GPU), reconstructions of sequences with several hundred frames can be generated in clinically useful time (less than 30 minutes). Details about the reconstruction method are explained in Section 5.1. The observed performance of the method as seen in several reconstructions is then shown in Section 5.2.

5.1 Methods and Materials

Reconstructing a 3DUS sequence to generate a high temporal resolution sequence is a multi-step process described in the following sections. The steps of the method are:

1. Resequence the original sequence according to the start time of each frame relative to the observed QRS peaks in the ECG (i.e. retrospective ECG-gating).

2. Stabilize the resequenced frames using an image-based rigid registration method.

3. Correct the temporal distortion in each frame of the resequenced and stabilized sequence.

5.1.1 Frame Resequencing

The frame resequencing method (retrospective ECG-gating) operates by taking the frames from a 3DUS acquisition of the heart over several cardiac cycles and reordering them so that all frames appear to have occurred during one cardiac cycle. This is done by recording both the start times of each frame and the ECG signal during the acquisition . The peaks of each QRS complex in the ECG signal are then determined, and the time offset

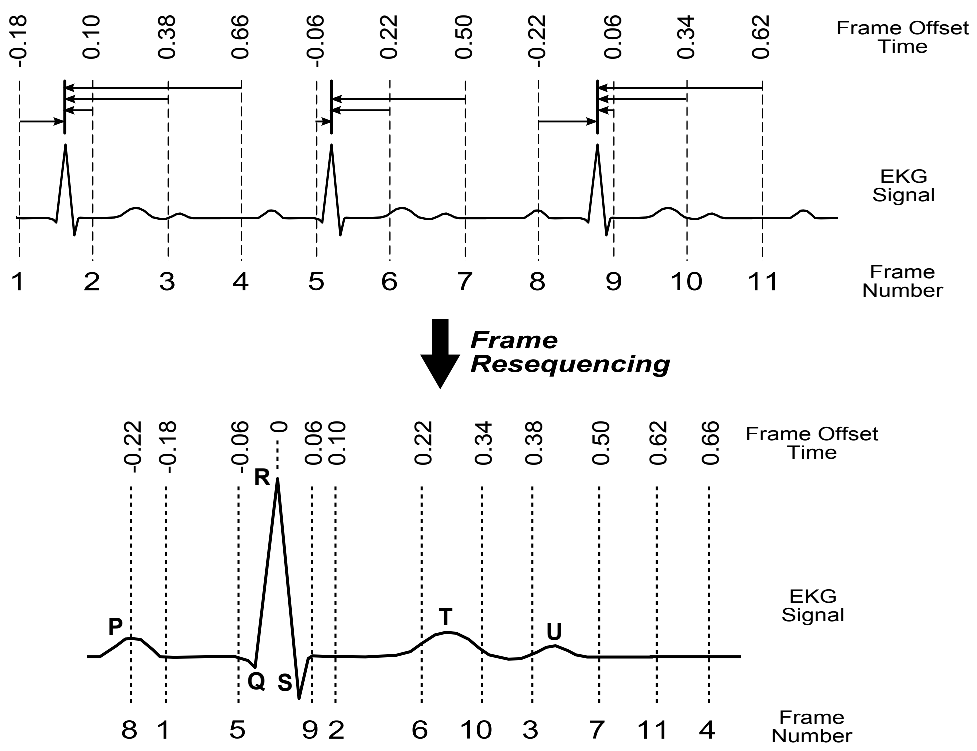


Figure 5.3: Example of the frame resequencing method. The top row shows the start time of each frame (which are 0.28 seconds apart) in the original acquisition relative to the ECG signal, and shows the temporal offset (assuming a heart rate of 60 beats per minute) of each frame from the R-peak to which they are associated. The bottom shows the resequenced order relative to a single cardiac cycle. Frames are resequenced according to their offset from the previous R-peak, unless the R-peak occurs during the frame acquisition (i.e. frames 1, 5, and 8), in which case the offset is from the R-peak that occurs during the frame.

of the start of each frame from the previously observed QRS peak computed. The frames are then ordered according to their offset from the QRS peaks (Figure 5.3).

The effective frame rate of the resequenced frames across a single cardiac cycle can be easily estimated based on information from the original acquisition. If n_f is the number of frames acquired in the original acquisition, d_f is the original frame rate (seconds/frame), and n_b is the number of cardiac cycles observed during the acquisition, then the average cardiac cycle time is

$$d_b = \frac{n_f d_f}{n_b} \tag{5.1}$$

Because the number of frames resequenced across the cardiac cycle is the same as the number of frames in the original acquisition, the resulting average resequenced frame rate (seconds/frame) is

$$d_{f,avg} = \frac{d_b}{n_f} = \frac{n_f d_f}{n_f n_b} = \frac{d_f}{n_b} \tag{5.2}$$

For example, if the original frame rate is 8Hz (0.125 seconds/frame) and 80 cardiac cycles were observed during an acquisition, the average resequenced frame rate is $d_{f,avg} = \frac{0.125}{80} =$ 1.5625 milliseconds/frame (640Hz).

While we can estimate an average resequenced frame rate using information about the ultrasound acquisition and heart rate, this is not to say that an even and adequate sampling across the cardiac cycle always results. In fact, when using retrospective ECG-gating, variability in the time between resequenced frames is inevitable mostly due to small changes in heart rate. It is therefore useful to determine when adequate or inadequate coverage across the cardiac cycle is expected. For this purpose, we estimate the relationship between the frame rate and heart rate as

$$\alpha d_b = \beta d_f + \phi \tag{5.3}$$

where d_b is the average heart rate, d_f is the average frame rate of the original acquisition, α and β are positive integers, and ϕ is a positive scalar. This relationship is assuming that over the course of α cardiac cycles, β frames are acquired, and the temporal remainder is ϕ. The integers α and β are found as the smallest possible integers such that $\phi < d_r$, where d_r is the desired frame rate or maximum allowable time between resequenced frames as decided by the user. It therefore follows that ϕ is a phase shift that accumulates after every β frames, and the total phase shift that accumulates during an original acquisition is $\Phi = \frac{\phi n_f}{\beta}$, where n_f is the number of frames in the original acquisition. For adequate coverage across the cardiac cycle to result, the inequality that must hold is $\Phi > \frac{d_b}{\beta}$. This can also be interpreted as $n_f > \frac{d_b}{\alpha d_b - \beta d_f}$ or $d_f < \frac{d_b(n-1)}{\beta n_f}$. This shows that for good coverage across the cardiac cycle, the original frame acquisition time should be made as small as possible and a large number of frames should be acquired.

The prediction of adequate sampling across the cardiac cycle relies on the assumption of a constant heart rate. While the heart rate can change in a stochastic fashion, and while this change could possibly be modeled using statistical analysis, we found that the change that did occur was not substantial and did not greatly alter the predictions. Furthermore, because the acquisitions used for the reconstruction pipeline did not take a long time (most acquisitions took about 45 seconds) and because patients were not required to cease respiration (which can affect heart rate), the resulting heart rate was fairly constant.

5.1.2 Image Stabilization

The resequenced ultrasound images are originally acquired in 30–60 seconds, depending on the volume size and ultrasound machine settings. The acquisitions are generally taken using either a transesophageal approach (TEE), in which case the patient is sedated and the ultrasound probe is placed in the esophagus and positioned to image the heart, or a transthoracic approach (TTE), in which case the patient is either awake or sedated and the heart imaged with the ultrasound probe outside the body. In both approaches, the patient is allowed to breathe, as holding respiration for 30–60 seconds can place unnecessary strain on patients. Therefore, because the position of the heart relative to the ultrasound probe is affected by the changing lung volume, the heart appears to move relative to the probe in the acquired ultrasound image sequence. Additionally, in TTE cases where the patient is not sedated, the patient is likely to move during the acquisition, which again would cause the heart to appear to move in the image sequence. This is especially true in pediatric cases, as children are typically more restless than adults.

These shifts (in rotation and/or displacement) of the heart in the image sequence need to be resolved before the temporal distortion and incoherence in the resequenced images can be addressed, as the methods that resolve these issues, which are described in the following sections, require that the same anatomy reside in the same region in neighboring images. As an effort is made during the acquisition to keep the probe stationary relative to the cardiac anatomy, these shifts are generally small compared to the size of the 3DUS volume, and are accounted for using the fast image-based rigid registration algorithm described in Chapter 6. This method spatially transforms the volumes in the sequence and negates any small motions that might have occurred.

5.1.3 Temporal Map Generation

Resequencing frames from an acquisition covering several cardiac cycles results in a sequence over a single cardiac cycle where start times of frames are just milliseconds apart. However, the volumes in each frame of the original sequence were still acquired over tens to hundreds of milliseconds, which means in the resequenced frame order, there is information in a frame that occurs *after* the start of the next frame in a resequence. It is therefore necessary to resolve this incoherence such that information shown in a frame is from a time between the start of the frame and the start of the next frame. Resolving this incoherence requires the approximate time at which the data in a volume was acquired relative to the start of the volume acquisition, which we will refer to as a voxel timestamp.

A 3DUS volume is a compilation of scan lines that are acquired in an organized manner (Figure 5.4). The shape of the acquired volume (i.e. locations of scan lines), scan line density, and order in which scan lines are acquired in a volume vary depending on the 3DUS machine manufacturer and machine settings. Fortunately, the information needed to determine an approximate voxel timestamp can be determined through experimentation and does not require proprietary information.

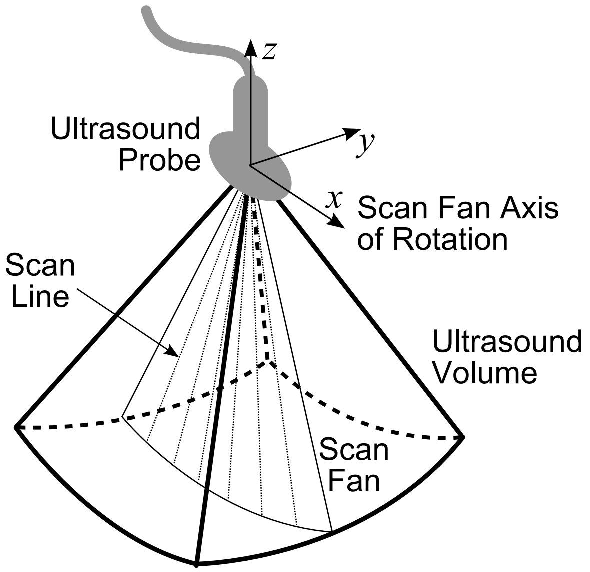


Figure 5.4: Depiction of location and orientation of scan lines and scan fans in a 3D ultrasound volume.

For 3DUS machines that perform single beam processing, generally speaking, scan lines are acquired in scan fans, which are similar to conventional 2D ultrasound images. Each fan contains a fixed number of scan lines, and each volume a fixed number of fans. The number of scan lines per fan (s_f) and fans per volume (f_v) is typically proprietary information, although a rough approximation for these values can be made [8]. The speed of sound in human tissue is roughly 1540m/s. If the volume depth is d_{vol}, then the amount of time needed to acquire each scan line is

$$t_{\mathrm{line}} = \frac{2d_{\mathrm{vol}}}{1540} \tag{5.4}$$

where the factor of two comes from the fact that the sound needs to travel down the specified depth and back up to the probe. Therefore, if it takes roughly t_{vol} seconds to acquire a volume, the number of scan lines that make up the entire volume is

$$n_{\mathrm{lines}} = \frac{t_{\mathrm{vol}}}{t_{\mathrm{line}}}. \tag{5.5}$$

If we assume equal volume dimensions along each side of the pyramidal-shaped volume, then

$$s_f \approx f_v \approx \sqrt{n_{\text{lines}}} \tag{5.6}$$

although approximations for s_f and f_v can be adjusted depending on the ratio of volume dimensions. As an example, if we acquire a sequence with a TEE probe using a depth of 8cm, and the specified volume size results in a frame rate of 20Hz, then $t_{\text{line}} \approx 104\mu s$, $n_{\text{lines}} \approx 481$ lines, and $s_f \approx v_f \approx 22$.

The s_f and f_v approximations are important because they provide the basis on which simplifying assumptions about the scan line order can be made. Because details of the scan line formation process in a volume is not known, the best generic approximation about the formation process that can be made is to assume a smooth temporal profile from one side of the volume to the other (Figure 5.5). A method to determine which side is acquired first and which last is discussed in Section 5.2.1. We can make this approximation because we can assume that the time to acquire all scan lines in a fan is much less (i.e. an order of magnitude) than than the time it takes, for instance, for a valve to close. For instance, in the reconstruction studies that we conducted and describe in Section 5.2.2, we found that the time for a valve to open or close was roughly 20-40 milliseconds, whereas the time to acquire the scan lines in a fan was roughly 2 milliseconds.

While the data is acquired as a compilation of scan lines and fans in a spherical coordinate space centered about the ultrasound probe, the image data is commonly in the form of a Cartesian grid, where the scan line data has been interpolated at each grid location. The grid locations are voxels in the 3D volume. At voxel locations corresponding to the region outside the conical data volume swept out by the scan lines, the value stored at the voxel is zero. It therefore follows that voxel timestamps are only defined for those voxels at the location of the conical data volume, and are not defined for those in the zero-padded regions.

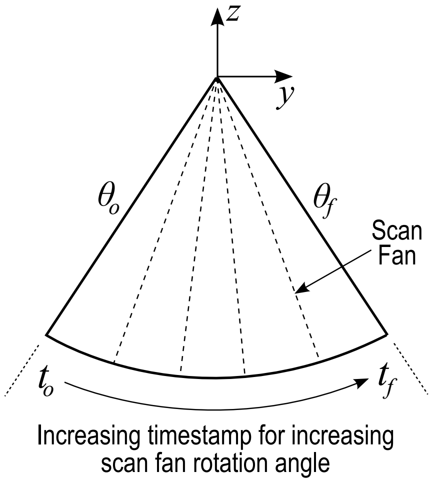
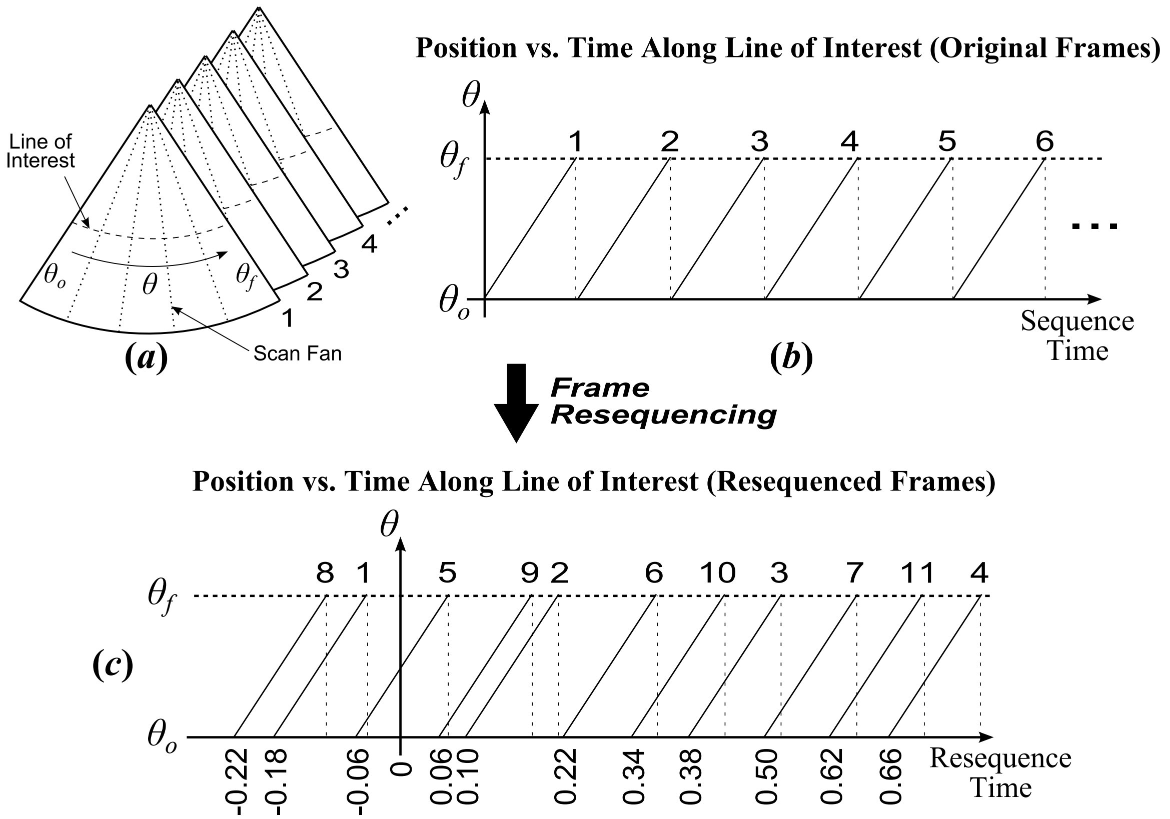


Figure 5.5: Summary of assumed temporal profile for a volume in a 3DUS acquisition. This assumption is valid for volumes acquired on most 3DUS machines that perform single beam processing.

By making the assumption of a smooth temporal profile from one side of the volume to the opposite, we can assign voxel timestamps based on their angular position within the volume. The angular position of voxel i in frame j is defined as its rotation angle, $\mathbf{\Theta}_{ij}$, about an axis defined as the line of intersection of the planes containing the first and last fans in the 3DUS volume, with $\mathbf{\Theta}$ representing the position profile of an entire volume. If the angles for the planes containing the first and last fans are θ_o and θ_f, respectively, and the start and end time for the acquisition of frame j are t_{oj} and t_{fj}, respectively, then the voxel timestamp for voxel i in frame j is

$$\mathbf{T}_{ij} = \left(\frac{\mathbf{\Theta}_{ij} - \theta_o}{\theta_f - \theta_o} \right) (t_{fj} - t_{oj}) + t_{oj} \tag{5.7}$$

where $\mathbf{T}$ represents the temporal profile of the volume. The voxel timestamps are absolute times that each voxel is estimated to have been acquired within the cardiac cycle. At voxels outside of the conical data region (i.e. in the zero-padded region) neither $\mathbf{\Theta}$ or $\mathbf{T}$ are defined.


Position vs. Time Along Line of Interest (Original Frames)

Position vs. Time Along Line of Interest (Resequenced Frames)

Figure 5.6: Simplified depiction of temporal incoherence of resequenced frames using the sequence data shown in Figure 5.3. (a) Simplified 2D cut planes from an original 3DUS sequence showing a line of interest in the volumes. (b) Example of position versus time along the line of interest for the native 3DUS sequence shown in Figure 5.3, which had a frame acquisition time of 0.28 seconds. Notice that in the original sequence, there is no temporal overlap of data along the line of interest. (c) Example of position versus time along the line of interest for the resequenced order. Notice, for example, the temporal overlap of resequenced frames 8 and 1. This indicates that in the resequenced frames the end of frame 8 occurs *after* the start of frame 1. This temporal incoherence problem will occur when the inter-frame time of the resequenced 3DUS frames is less than the time to acquire the original frames.

5.1.4 Frame Reconstruction

The temporal order in the resequenced frames is often such that the end of one frame will occur after the start of the next frame in the resequence. This is depicted in a simplified example in Figure 5.6. We correct this temporal incoherence problem using either a temporal sorting method (Section 5.1.4) or temporal weighting method (Section 5.1.4). These are voxel-based methods that use information from several of the resequenced frames to determine the most appropriate intensity value for a particular voxel in a given frame. While we chose to reconstruct frames at the start times of the frames in the resequenced acquisition, frames can be reconstructed at any arbitrary time within the cardiac cycle.

Temporal Sorting

The temporal sorting method operates independently at each voxel location in the reconstructed sequence to find a new intensity value at each voxel location, $\mathbf{I}_{ik}$, where i indicates the voxel location in the volume and k indicates the frame. The objective is to find for each voxel the frame j_{target} whose voxel timestamp is closest to the time at which a frame is to be reconstructed. It follows then that if the start time of frame k is t_{ok}, then for voxel location i,

$$j_{\text{target}} = \underset{j}{\operatorname{argmin}} \, |t_{ok} - \mathbf{T}_{ij}| \tag{5.8}$$

$$\mathbf{I}_{ik} = \mathbf{I}_{ij_{\text{target}}} \tag{5.9}$$

This means that out of all frames in the resequence, voxel i in frame j_{target} is temporally closest to the start time of frame k, and so the intensity $\mathbf{I}_{ik}$ is replaced by the intensity $\mathbf{I}_{ij_{\text{target}}}$. In this way, voxel intensities are merely shuffled among the frames and not recomputed.

An example of how different frames contribute to different locations in a frame is shown in Figure 5.7 for the case shown in Figure 5.6. This method results in images with

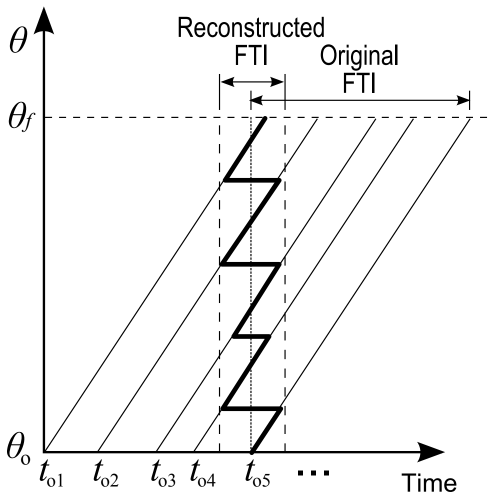


Figure 5.7: Example showing where data along a line of interest (for example, see Figure 5.6(a)) for a generic series of resequenced frames would originate for the temporal sort method (bold line) when reconstructing the frame at t_{o5}. Notice that the resulting FTI is less than the original FTI.

discontinuities due to the temporal discontinuities that occur across the volume. This is shown in the experiments described in Section 5.2.2.

Temporal Weighting

As opposed to performing a temporal sorting of the intensities from each frame, the temporal incoherence problem in the resequenced 3DUS frames can alternatively be remedied using a voxel-based temporal weighting method. In this method, the intensity at a voxel location is

$$\mathbf{I}_{ik} = \mathbf{N}_{ik} \sum_{j=1}^{n_f} \mathbf{W}_{ij} \mathbf{I}_{ij} \tag{5.10}$$

where

$$\mathbf{N}_{ik} = \frac{1}{\sum_{j=1}^{n_f} \mathbf{W}_{ij}} \tag{5.11}$$

$$\mathbf{W}_{ij} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp^{-\frac{\left(\mathbf{T}_{ij} - t_{ok}\right)^2}{2\sigma^2}} \tag{5.12}$$

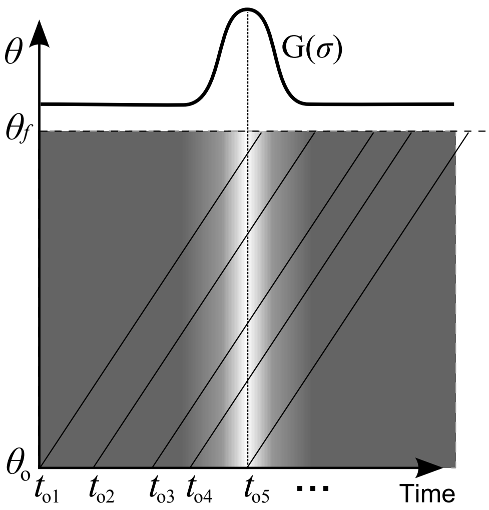


Figure 5.8: Example showing weighting of data along a line of interest (for example, see Figure 5.6(a)) for a generic series of resequenced frames. The weighting shown is for reconstructing a frame at t_{o5} using the temporal weighting reconstruction method. Data in the lighter regions contribute more to the image at the corresponding location than data in the darker regions. The plot on the top shows the Gaussian weighting function relative to the frame time.

The normalization factor, $\mathbf{N}_{ik}$ is necessary to ensure the intensity is not undesirably attenuated. $\mathbf{W}_{ij}$ is a Gaussian weighting of the intensities at the corresponding voxel location from all other frames based on their temporal offset from the start of the frame (t_ok), where n_f is the number of frames. The standard deviation of the Gaussian for the weighting distribution is

$$\sigma = \frac{1}{n_f - 1} \sum_{k=2}^{n_f} t_{ok} - t_{o(k-1)} \tag{5.13}$$

which is the average time difference between resequenced frames.

A depiction of the weighting distribution relative to the resequenced frame data is shown in Figure 5.8. The data in the lighter region will contribute more to the voxel at the corresponding location than the data in the darker region. While this method generates images that are smoother in nature than the temporal sort method, due to the nature of the weighting method, a ghosting effect is possible.

5.2 Experiments and Results

5.2.1 Determining Scan Fan Rotation Axis in an Ultrasound Volume

The reconstruction algorithm operates on the assumption that the conical 3DUS volume is acquired by rotating the scan fan from one side of the cone to the other, thereby resulting in a smooth temporal profile across the volume. This is a reasonable assumption to make for machines that perform single beam processing. It is therefore important to be able to determine the scan fan rotation axis so that an accurate temporal profile can be generated.

The rotation axis was found through a simple experiment involving the ultrasound probe of interest and a water tank. The probe was supported such that the bottom of the tank was visible in the volume. During the acquisition of a 3DUS sequence, the probe was swiftly displaced a short distance toward the bottom of the tank and then held stationary, all the while keeping the probe perpendicular to the bottom of the tank. The idea was that in the frames where the probe was displaced, the bottom of the tank would appear curved in the direction of volume acquisition (i.e. between scan fans) and flat in the opposite direction (i.e. within a scan fan). The size of the acquired 3DUS volume using an iE33 Echocardiography System with a transthoracic X7-2 probe (Philips Healthcare, Andover, MA, USA) was made to be very large ($320 \times 320 \times 208$ voxels with a resolution of $0.4331 \times 0.4331 \times 0.4837$ mm/voxel) so that a small frame rate would result (about 4Hz), making the frame acquisition time very long (250 milliseconds). This resulted in a large distortion and made it easier to determine the direction of acquisition within the volume.

The results of this experiment can be seen in Figure 5.9. Before the probe was displaced, the bottom of the tank appears flat in both the xz and yz cut planes. However, during the time when the probe was displaced, the bottom of the tank appears curved in

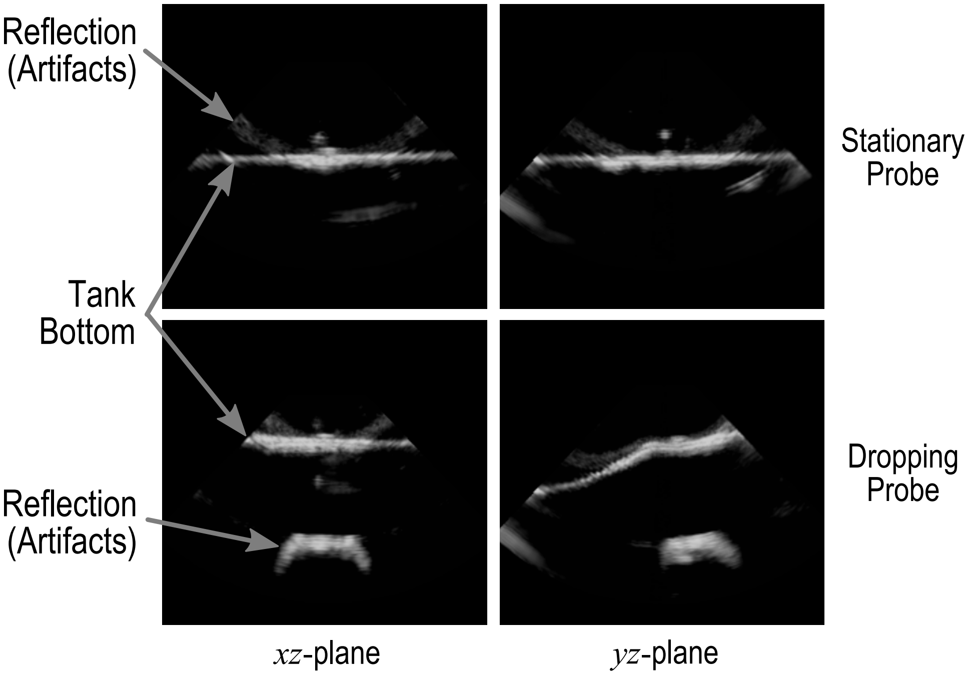


Figure 5.9: Cross sectional planes showing the bottom of the water tank while the probe was suspended and while the probe was displaced toward the bottom of the tank. The fact that the bottom of the tank appears flat in the xz–plane but curved in the yz–plane while the probe is displaced indicates the direction of acquisition of the ultrasound volume is about the x–axis.

the yz–plane, indicating that the direction of acquisition within the ultrasound volume is about the x axis.

While we determined the direction of acquisition through experimentation in a water tank, this direction can also be determined through careful examination of clinical data. This is done by noting the location of scan fans, which are most visible when they try to capture fast moving thin structures, such as heart valve leaflets. For instance, in Figure 5.10, which is a clinical TEE image of a mitral valve acquired at 24Hz, the location of the scan fans are apparent by noting the striped appearance of the anterior mitral leaflet. The stripes, which are oriented in the x-direction, are the intersection of the scan fans with the leaflet, and indicate that the rotation axis for the scan fans is parallel to the x-axis.

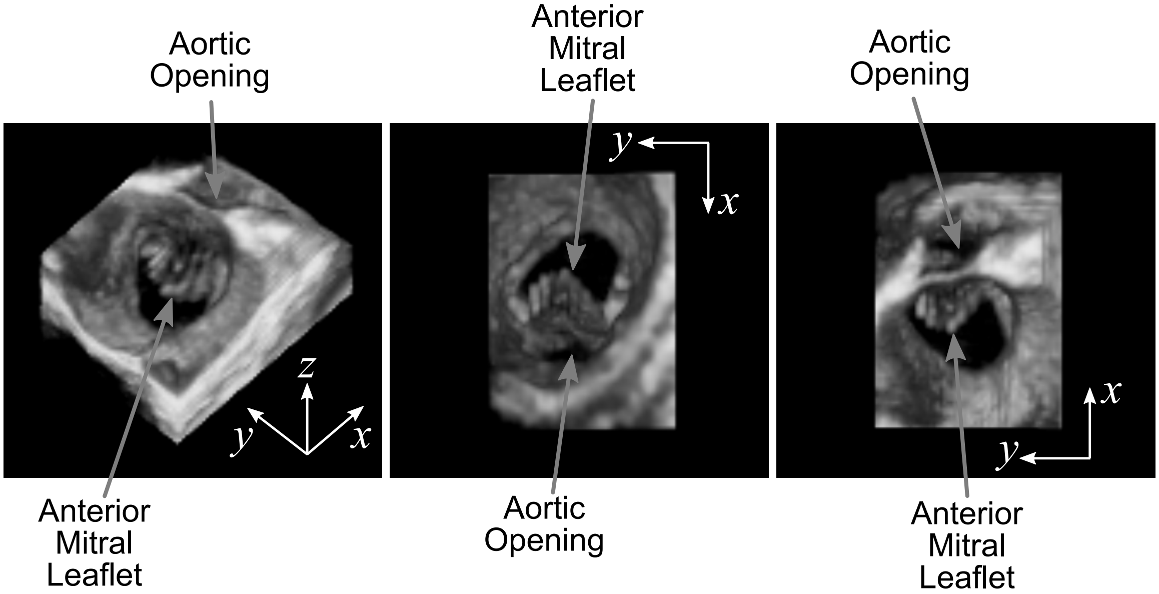


Figure 5.10: Clinical ultrasound of a mitral valve which showing the location of the scan fans at the anterior mitral leaflet in (*L*eft) an isometric view from above the valve, (*M*iddle), a view looking up at the valve from the left ventricle, and (*R*ight) a view looking down on the valve from the left atrium. The locations of the scan fans are characterized by the striped appearance of the anterior mitral leaflet. The fact that the stripes are directed in the x-direction indicates that the scan fans are rotated about an axis parallel to the x-axis.

5.2.2 Reconstruction Results

We have used the 3DUS reconstruction pipeline in several clinical cases, a summary of which appears in Table 5.1. The cases involved either the aortic or mitral valve, and while most were acquired using a TEE approach (X7-2t probe, iE33 Echocardiography System, Philips Healthcare, Andover, MA, USA), studies 24–27 were acquired using a TTE approach (X7-2 probe). Several of these reconstructions have been used by cardiologists and cardiac surgeons in preoperative planning.

The original acquisitions took an average of 38 seconds and were acquired at an average frame rate of 9Hz. The length of the acquisition was not specified but rather was determined by the maximum sequence length that the 3DUS machine could process and store for the given volume size. The low frame rate was a result of using a large volume,

Table 5.1: Summary of Original and Reconstructed 3D Ultrasound Acquisitions. The original and reconstructed frame rate and FTI are observed averages. Data sets 1-23 were acquired using a transesophageal approach and 24-27 using a transthoracic approach.

	Acquisition			Original		Reconstructed	
Data	Duration		Cardiac	Frame	FTI	Frame	FTI
Set	(seconds)	Frames	Cycles	Rate (Hz)	(ms)	Rate (Hz)	(ms)
1	45	302	81	6.7	149.3	543.5	1.8
2	34	353	60	10.4	96.4	556.4	1.8
3	37	367	64	9.9	100.8	620.8	1.6
4	30	435	54	14.5	69.0	780.7	1.3
5	49	323	88	6.6	152.1	578.9	1.7
6	23	385	41	16.7	59.9	673.9	1.5
7	50	242	59	4.8	207.1	281.4	3.6
8	43	236	51	5.5	182.5	273.5	3.7
9	42	277	62	6.6	152.1	394.6	2.5
10	41	302	55	7.4	135.9	404.1	2.5
11	46	242	54	5.2	190.6	283.2	3.5
12	43	239	49	5.5	180.5	268.0	3.7
13	29	352	42	12.1	82.5	495.4	2.0
14	29	352	41	12.1	82.5	494.6	2.0
15	40	273	66	6.8	147.0	446.5	2.2
16	44	242	73	5.5	182.5	399.2	2.5
17	27	448	38	16.6	60.3	615.7	1.6
18	38	310	55	8.1	122.8	434.1	2.3
19	24	435	38	18.1	55.3	627.5	1.6
20	36	303	49	8.4	118.9	409.7	2.4
21	34	306	29	9.0	111.2	249.2	4.0
22	20	196	23	9.8	102.4	216.2	4.6
23	45	247	70	5.5	182.5	383.4	2.6
24	37	266	59	7.2	139.4	401.7	2.5
25	32	505	49	15.8	63.4	685.5	1.5
26	55	372	54	6.8	147.9	326.0	3.1
27	60	311	81	5.2	193.1	422.7	2.4
Average	38	319	55	9.1	128.4	454.3	2.5
Std. Dev.	9.6	74.6	15	4.0	46.2	148.0	0.9

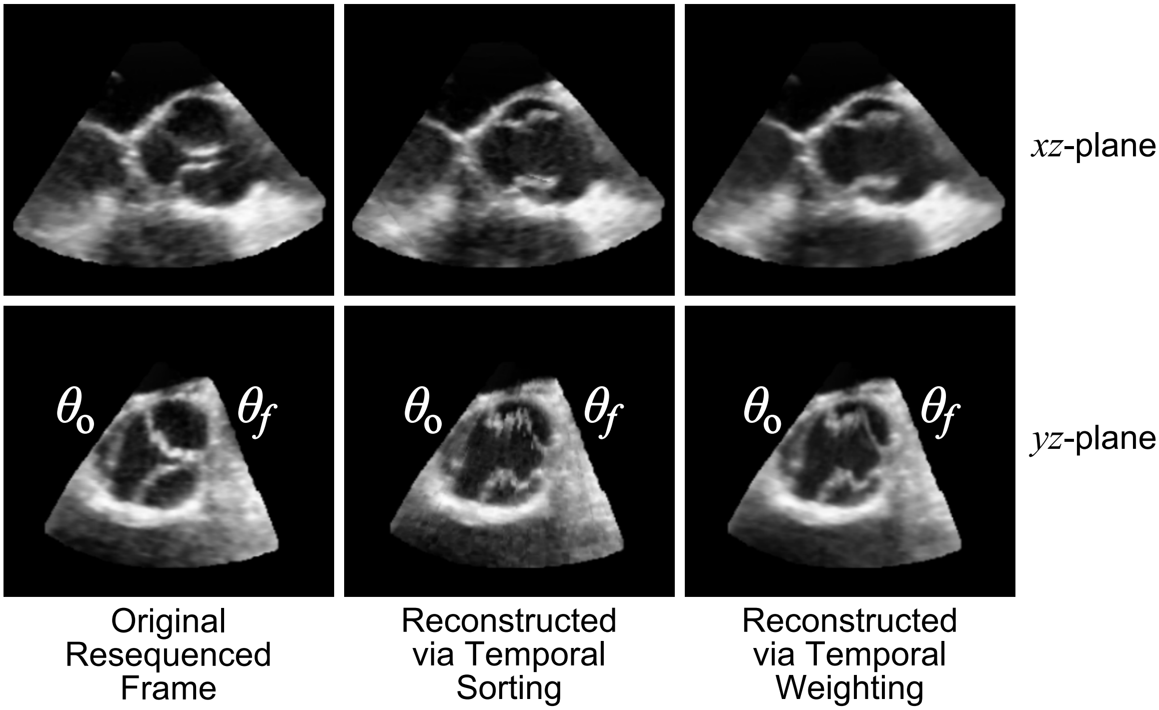

xz-plane
yz-plane
θ_o
θ_f
θ_o
θ_f
θ_o
θ_f
Original Resequenced Frame
Reconstructed via Temporal Sorting
Reconstructed via Temporal Weighting

Figure 5.11: Orthogonal planes from three 3DUS volumes of an aortic valve comparing (Left) a native resequenced frame, (Middle) a frame reconstructed using temporal sorting, and (Right) a frame reconstructed using temporal weighting. All frames have the same "start" time at θ_o, which explains why they have a similar appearance near θ_o. However, due to temporal distortion, the frame on the left appears to be at a different phase in the cardiac cycle compared to the middle and right frames. Using the sorting method, discontinuities in the reconstructed frames are visible, while the weighting method displays a small degree of ghosting, or lack of definition. In these frames, the volumes were acquired by rotating the scan fans about an axis parallel to the x–axis located at the top of the cone.

which was necessary to capture the extent of the aortic or mitral valves. In both the TEE and TTE acquisitions, the patients were *not* put on breath hold. The resulting reconstructions had an average frame rate of 454Hz, which meant the FTI for the reconstructed frames was only about 2.5ms. The reconstruction pipeline was made to be fully automated, and the average reconstruction time for the studies was on the order of 20 minutes using a GPU accelerated architecture.

An example of a frame reconstructed using the temporal sorting versus using the temporal weighting approach is shown in Figure 5.11. As is visible and previously mentioned, the sorting method is prone to discontinuities in the reconstruction, while the weighting method is prone to ghosting effects due to the inclusion of information at a voxel from times that are farther away from the target frame time. With both methods, it was observed that these undesired characteristics were minimized in reconstructions with a low FTI.

5.3 Discussion

5.3.1 Reconstruction Performance

We have shown that the 3DUS reconstruction pipeline is useful for improving the temporal resolution of a standard 3DUS acquisition. This is done by first resequencing the frames according to their position relative to the QRS complex in the ECG. Then, by assuming a temporal profile within each frame, the intensities at each voxel can be found either by performing temporal sorting or temporal weighting. The result was that for an acquisition of about 40 seconds, the temporal resolution was improved by a factor of 50 (or the number of cardiac cycles observed in the acquisition).

It is important to note that the reconstructions were generated from clinical data that was acquired with minimal deviation from current clinical practice. The only change from current clinical imaging practices was the length of the acquisition, which was on the order of 40 seconds, whereas standard images usually take only a few seconds. However, no additional hardware or data acquisition was required. Additionally, because we compensated for small displacements frame to frame, we did not have to put patients on breath-hold (or perform respiration gating). This allowed the necessary 3DUS sequences to be acquired with little strain on the patients.

The reconstructions were performed on data that was acquired using both a transesophageal (TEE) and transthoracic (TTE) approach. That being said, it was found that automated processing of TEE acquisitions was easier only because in TTE acquisitions, there were instances in which the probe would lift off the patient, especially in cases involving non-sedated children. The acquisitions in these cases could be still be reconstructed, but the bad frames had to be removed so as not to adversely affect the image-based stabilization or the temporal distortion correction.

5.3.2 Assumptions and Limitations

While we have highlighted the fact that a temporal distortion exists within frames, in assuming that the distortion is constant among all frames and that frame-to-frame displacement is small, it is reasonable to assume that a rigid transformation between frames can be found using the image-based method described in the next chapter. It is interesting, however, to see how a displacement can affect not just the frame's position, but relative temporal location within the resequence. For instance, if we look at two consecutive frames in a resequence, if the time between the start of each frame is Δt and if the frames are aligned (frames A and B in Figure 5.12), then it stands to reason that at corresponding voxel positions, the voxels in the second frame will all occur after the first frame. However,

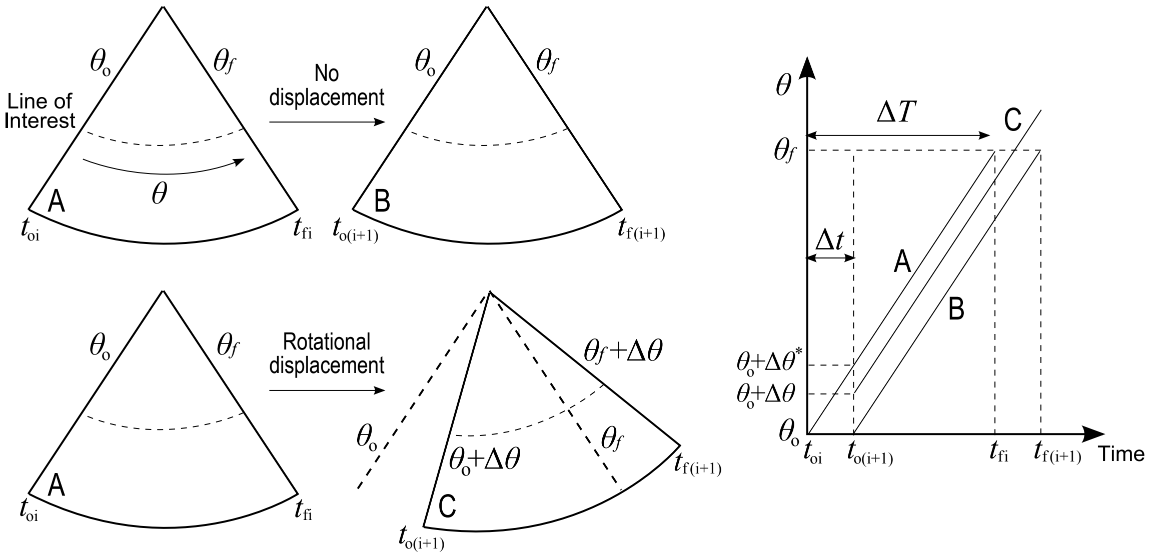


Figure 5.12: Example highlighting how rotational displacement and relative temporal location of voxels are related. The plot shows the time versus position for the line of interest shown in each frame. (Frames A-B) The frames are aligned, and so all voxels in the second frame occur after the first frame. (Frames A-C) Frame C is rotationally displaced such that the voxels in Frame C that overlap with those from Frame A have the potential to have occurred at or before the corresponding voxels from Frame A if $\Delta\theta \geq \Delta\theta^*$. This motivates the importance of temporal distortion correction in high temporal resolution reconstructions/acquisitions.

if the second frame is rotated relative to the first (frames A and C in Figure 5.12), then there is a rotational limit, $\Delta\theta^*$, at which the overlapping voxels in the second frame will have actually occurred before the corresponding voxels in the first frame. Using the parameters from Figure 5.12, this rotational limit is defined as

$$\Delta\theta^* = \frac{\theta_f - \theta_o}{\Delta T}\Delta_t \tag{5.14}$$

If the second frame is rotated in the opposite direction, the relative order of the voxels coincides with the order of the frames. This further motivates the importance of temporal distortion correction and registration in a high temporal resolution reconstruction. We should note that the small motion assumption was found to account for motion due to respiration in all studied cases highlighted in Table 5.1, which included both TEE and TTE

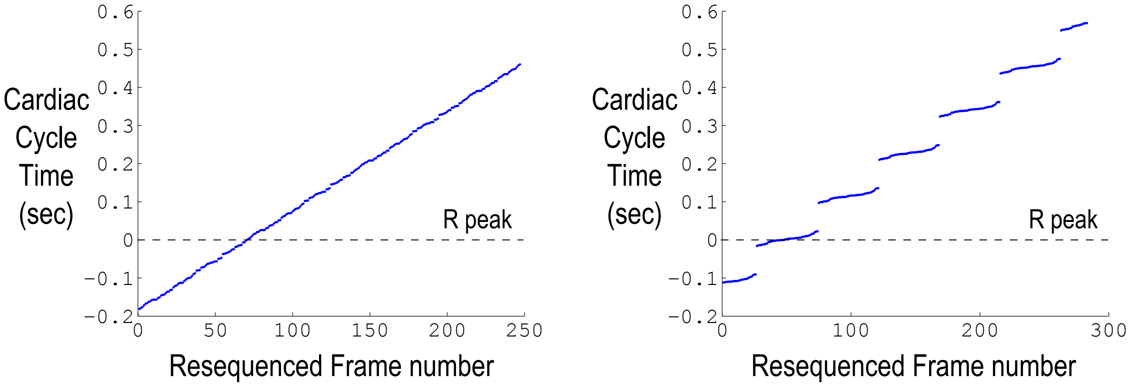


Figure 5.13: Examples which show (*L*eft) when resequencing frames results in adequate coverage over a cardiac cycle and reconstruction is successful and (*R*ight) when inadequate coverage over a cardiac cycle results and reconstruction fails. The latter occurred because the cardiac cycle time (0.677 seconds) was nearly six times the frame time (6×0.113 seconds $= 0.678$ seconds).

acquisitions.

The presented reconstruction pipeline operates on the assumption that over the length of the original acquisition, the acquired frames will occur at times during the cardiac cycle such that when the frames are resequenced, adequate coverage across the cardiac cycle is observed. This was the case for the 27 data sets summarized in Table 5.1. However, we did encounter two studies in which the frame rate and heart rate were such that inadequate coverage was not observed over the cardiac cycle, resulting in failed reconstructions. The failures occurred because the heart rate was steady and was nearly an integer multiple of the frame rate. The distribution of the frames across the cardiac cycle for a case where reconstruction is successful and for when reconstruction fails is shown in Figure 5.13. In the failure case shown, inadequate coverage resulted from the fact that the time it took to acquire six frames (6×0.113 seconds $= 0.678$ seconds) was nearly the same time for one cardiac cycle (0.677 seconds).

Currently the maximum length of the acquisition was limited by the amount of data that could be processed and stored by the 3DUS machine, which was about 4GB of data. The acquisition length was determined by the size of the volumes being acquired, which were made to be large enough to capture the extent of either the mitral or aortic valve. For the analyzed cases, this resulted in an original frame rate of about 10Hz and a maximum number of frames on the order of 300-400. Limiting the length of the acquisition in turn limited the number of observed cardiac cycles, which limited the temporal resolution improvement, as pointed out by (5.2). However, given that the STR and FTI for the reconstructions were on the order of 2ms, it was found that a much higher temporal resolution was not necessary for visualizing the aortic and mitral valves. Additionally, if a higher temporal resolution would be desired, because scan fans themselves took on the order of 2ms to be acquired, the temporal distortion within each scan fan would have to be resolved. This would likely require more (possibly proprietary) information other than just the assumption of a smooth temporal profile across the volume.

5.3.3 Future Work

The reconstruction pipeline is a fully automated method to improve the temporal resolution of a standard acquisition. However, as pointed out, there are a few limitations and undesirable characteristics which are to be addressed in our future work. As pointed out in the frame reconstruction section, the temporal sorting method and temporal weighting result in either discontinuities or ghosting effects, respectively. Future work will address the frame reconstruction to possibly incorporate a temporal interpolation of voxel intensity based on localized optical flow characteristics. Also, as highlighted in the discussion, there are instances in which the reconstruction fails due to inadequate coverage of the resequenced frames across the cardiac cycle. Future work will include developing real-time metrics by which failure cases can be detected and thereby avoided.

Chapter 6

Real-Time Image-Based Rigid Registration of 3D Ultrasound

To reconstruct high temporal resolution ultrasound from standard clinical acquisitions as described in the previous chapter, images need to be stabilized to account for respiration and other movements of the ultrasound probe relative to the anatomy of interest. This stabilization can be done using an image-based rigid registration algorithm under the assumption that small displacements occur from frame to frame. We developed such an algorithm originally for the purpose of creating large field-of-view (FOV) ultrasound mosaics. This chapter therefore describes our original problem of creating 3DUS mosaics and our proposed solution, along with a thorough validation of the designed method. We then show how the designed registration algorithm can be used for various applications, such as image stabilization.

Constructing large FOV 3DUS images from multiple smaller ultrasound frames allows for the visualization of expansive structures in a single volume. Before the availability of 2D array transducers, this was done primarily in the form of freehand 3DUS systems, where 2D ultrasound images are registered and mosaiced into arbitrarily large 3DUS volumes [70, 111, 88]. This is most commonly done by tracking the transducer with an external tracking system, usually optical or electromagnetic, and computing transformations between images based on the position and orientation of the tracker. These systems require careful calibration of the tracker to the ultrasound image [4, 5, 59]. Respiration also needs to be accounted for either by use of another tracker or by breath hold. Lastly, careful consideration of the tracking environment needs to be considered, as electromagnetic tracking is sensitive to metal in the field and optical trackers need to maintain a clear line of sight between the sensor and markers.

Even after the clinical availability of real-time 3DUS systems, efforts continue to construct larger FOV images by stitching together 3DUS volumes. Tracking-based systems, much like those used to generate 3D volumes from 2D ultrasound images, have been used to mosaic 3DUS volumes [71, 109, 114]. These systems are subject to the same limitations as their 2D counterparts, and so as with many freehand 3DUS systems, several image-based methods have additionally been used with tracker-based methods to more accurately register images [108, 24, 71, 109, 114]. However, whereas the intersection of neighboring 2D images in a freehand 3DUS acquisition is at most a single line, the intersection of neighboring 3DUS volumes in a mosaic is a smaller subvolume, allowing for the registration between 3DUS volumes using strictly image-based methods.

The strictly image-based 3DUS registration methods can be divided into two types: voxel-based and feature-based. Voxel-based methods compute a metric over all voxels in a volume (or overlapping voxels from volumes to be registered) and in an iterative fashion find the parameters of the transformation between volumes [77, 86, 23, 12, 65, 29, 101, 75, 48, 105]. These methods have proven to be accurate, even in the presence of large translational and rotational displacements between images, but because voxels are revisited several times, they can be slow. Fast implementations of this type [48, 105] used accelerated frameworks on a graphics processing unit (GPU) and simultaneously registered 3DUS volumes to each other and to CT images. The former registered larger volumes in 1-3 seconds, depending on volume size, whereas the latter registered 10 smaller intracardiac 3DUS volumes in 0.6 seconds by assuming a linear trajectory of the transducer and modifying only the first and last transformations.

The second image-based registration type consists of feature-based methods that compute a transformation between images by determining a correspondence between feature sets (i.e. segmented volume, edges, salient points, etc.) extracted from 3DUS volumes [73, 89, 63, 103, 67]. These methods can usually only handle small translational and rotational displacements between 3DUS volumes, but are generally faster than voxel-based methods. The fastest of these (*not* GPU accelerated) was shown in [67], which used a 3D SIFT implementation to register volumes in roughly one minute.

A limitation of current registration methods designed to generate large FOV 3DUS mosaics is that, because they do not operate in real-time, the extent of the mosaic cannot be immediately assessed, and so it is not known if sufficient or excessive coverage of the anatomy of interest has been obtained until well after the images are acquired. Additionally, the location of specific anatomic structures relative to the ultrasound probe cannot be obtained. This information would prove useful in interventional cases, such as when attempting to guide a needle to the location of a tumor.

To address these issues, we present an image-based 3DUS registration method capable of operating in real-time when using a GPU accelerated framework, thereby making large FOV 3DUS images and their evolution immediately available to clinicians. The method is a feature-based method, summarized in Figure 6.1. The presented method is designed for real-time ultrasound acquisitions that are acquired as continuous sweeps over

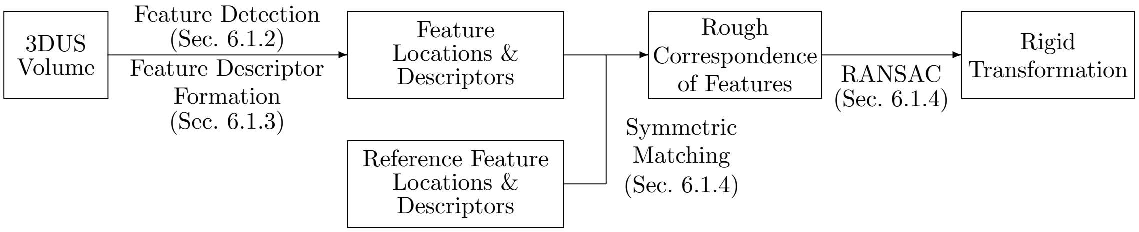

3DUS Volume
Feature Detection (Sec. 6.1.2)
Feature Descriptor Formation (Sec. 6.1.3)
Feature Locations & Descriptors
Reference Feature Locations & Descriptors
Symmetric Matching (Sec. 6.1.4)
Rough Correspondence of Features
RANSAC (Sec. 6.1.4)
Rigid Transformation

Figure 6.1: Summary of the feature-based 3DUS rigid registration algorithm. The specific processes can be found in the indicated sections.

the anatomy of interest, much like those used for freehand 3DUS generation, where small displacements occur between images in the sequence. This does not mean that large translations or rotations cannot occur over the course of the acquisition, but merely that from frame to frame these displacements are small. As probe movement in most acquisitions is already limited, such as on the surface of the liver or within the esophagus to view the heart, these restrictions should not limit the registration algorithm's usefulness. The method, however, is not designed to perform the registration between volumes that exhibit large translational or rotational displacements, such as used in several compositing studies [30, 109, 75].

This chapter is organized such that the details of the algorithm are outlined in Section 6.1. The accuracy, parameter selection, and execution times for the algorithm, using both clinical and water tank images, are then presented in Section 6.2.

6.1 Materials and Methods

6.1.1 Data Assumptions & Pre-Processing Strategies

This method is designed for real-time ultrasound acquisitions where data is streamed from the ultrasound machine to an external CPU. Using both the CPU and an accompanying GPU, transformations and interpolations are computed. To expedite the processing of the 3DUS volumes, we exploit several characteristics of ultrasound sequence data. First, the number of intermediate volumes needed to process each volume before it can be added to a mosaic is known. Therefore, to avoid the dynamic allocation/deallocation of memory, all necessary volumes are initialized and reused as necessary. Second, there are several intermediates, namely eroded data masks that are used to eliminate edge effects at the conical data boundary, that are the same for every volume in a sequence. We therefore precompute and store these intermediates so that they can be quickly queried at a later time. Lastly, there are some computations that need to be performed at each voxel, however, in the zero-padded region surrounding the conical data volume, these operations are meaningless. Therefore, the number and locations of the conical data voxels are stored so that efficient thread launches on the GPU can be executed to perform these computations.

It is worth noting that with the CUDA and device architecture used in this study (NVIDIA GTX 260 Graphics Processing Unit, Driver/Runtime Version 3.0, Compute Capability 1.1, NVIDIA Corporation, Santa Clara, CA, USA), the time to load a 3DUS volume ($\sim 128^3$) onto the GPU as a texture took about 20 times longer than loading the volume onto the GPU as a linear memory block. Therefore, linear memory blocks in global memory were used for all data loaded onto the GPU in this study.

6.1.2 Feature Detection

Several methods can be used to find feature points in an image [61], but the most important characteristic of the feature detection is that the feature points are stable (i.e. features are found at the same salient region) from image to image. In the 2D SIFT implementation for photos [56] and 3D SIFT implementation for 3DUS [67], these features are found as the local extrema of a Difference-of-Gaussian (DoG) scale-space. While this is important for photos, as the scale of objects can change according to the orientation of the camera relative to an object, this is unnecessary for 3DUS, as objects always appear at the same scale.

We therefore simplify feature detection by searching for local minima of the Laplacian-of-Gaussian (LoG) of a 3DUS volume at a single user-specified scale, σ_f,

$$\begin{aligned} LoG &= \nabla^2 G_{\sigma_f}(x,y,z) \\ &= \frac{x^2+y^2+z^2-3{\sigma_f}^2}{{\sigma_f}^5\sqrt{2\pi}} \exp\left[-\frac{x^2+y^2+z^2}{2{\sigma_f}^2}\right] \qquad (6.1) \\ G_{\sigma_f}(x,y,z) &= \frac{1}{\sqrt{2\pi{\sigma_f}^2}} \exp\left[-\frac{x^2+y^2+z^2}{2{\sigma_f}^2}\right]. \end{aligned}$$

We use the LoG as opposed to the DoG because the DoG is just an approximation to the scale-normalized LoG [56], which was shown in [61] to produce the most stable image features compared to other image functions. We search only for local minima of the LoG (using a 27 voxel neighborhood), as opposed to both local minima and maxima, because we are only interested in finding features at probable tissue locations, and want to avoid finding features in the middle of blood pools or shadows that appear in a 3DUS image. To further enforce that features reside at tissue locations, we require that the 3DUS intensity at a feature location is above a user-specified tissue threshold, τ_{tissue}. An important trait of this feature detection method is that, while this does not find all features at all spatial scales, because features do not change scale in 3DUS, the same features at a constant scale are being found from image to image. Assuming an appropriately chosen value for the scale, σ_f, a sufficient number of features (on the order of several hundred features) should be found such that an accurate registration between images can be computed.

The intent of having the user define σ_f and τ_{tissue} is to allow the user some control over the number of features that are found in the images, and to allow the user to impart prior knowledge about the expected location of salient regions in the image. It stands to reason that for increasingly smaller scales or tissue thresholds, more features will be found.

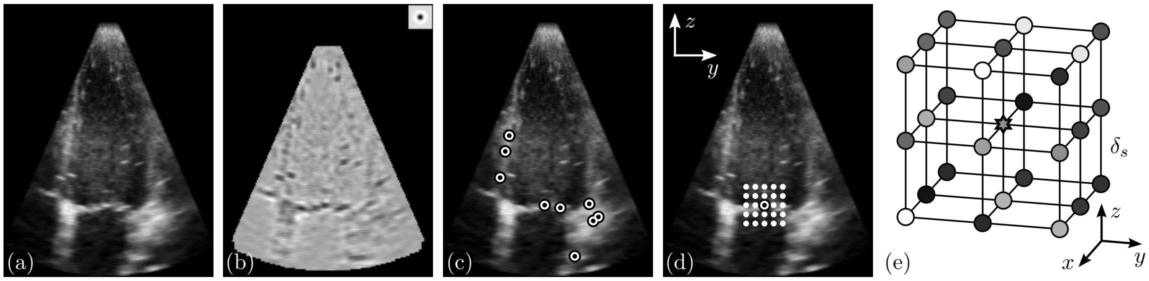


Figure 6.2: The feature detection and feature descriptor formation process. (a) Slice from a clinical 3DUS image of the heart. (b) Corresponding slice from the Laplacian-of-Gaussian (LoG) computed for the original volume shown in (a) (σ_f = 2.0mm). The image in the top right corner of (b) shows the center slice of the LoG kernel (to scale) corresponding to σ_f. (c) Same slice as (a) with detected feature locations found as the local minima of the volumetric LoG and with intensity greater than $\tau_{tissue} = 150$. (d) 2D depiction of sample points about a feature to construct the feature descriptor. (e) Example of points sampled in 3D relative to the feature point (starred) to construct the feature descriptor. The shaded spheres represent the sampled intensities of the 3DUS volume. The grid axis is aligned with the example image axis shown in (d). Note that while a 3×3×3 grid is shown, the actual grid used to compute feature descriptors is a 5×5×5 grid with a sample offset of δ_s.

Typical values for σ_f and τ_{tissue} used in studies were 1-2mm and 150-200, respectively, where 3DUS intensities were in the range of 0 to 255.

6.1.3 Feature Descriptors

A feature descriptor is a means of uniquely characterizing a feature location and allows for the calculation of a correspondence between features from different images. Previous efforts have been made to make these descriptors both scale and rotationally invariant in both 2D and 3D imaging for the purpose of matching features under large rotations and changes in scale [56, 67]. However, when registering volumes in a real-time 3DUS sequence, the rotations from frame to frame are small. We therefore construct feature descriptors using a simple rotationally variant method.

We construct feature descriptors by taking a sparse sampling of the 3DUS volume at and around a feature location (Figure 6.2). The sampling is taken on a rectilinear $5 \times 5 \times 5$

grid of 125 points, where the grid is oriented along the image axes and centered at each feature location. The sample spacing, δ_s, is equal to $M_d \sigma_f$, where the scale factor M_d is a constant and ensures that the extent of the feature is represented. The samples are then organized as feature vectors of dimension 125 normalized to unit norm. We show in several validation studies that these descriptors provide a sufficient level of performance with regard to registration accuracy, and additionally take little time to compute.

6.1.4 3DUS Volume Registration

Volume registration is performed using the feature sets from the volumes to be registered. In this study, we refer to the reference (stationary) volume as $\mathbf{V}_{ref}$ and its features as $\mathbf{F}_{ref} = (\mathbf{X}_{ref}, \mathbf{D}_{ref})$, where $\mathbf{X}_{ref}$ and $\mathbf{D}_{ref}$ are the positions and feature descriptor vectors, respectively, of the reference feature set. The number of features in the set is n_{ref}, and the position and feature vector for feature i in the set is $\mathbf{X}_{i,ref}$ and $\mathbf{D}_{i,ref}$, respectively. Similarly, we refer to the volume to be registered as $\mathbf{V}_{new}$ and its features as $\mathbf{F}_{new} = (\mathbf{X}_{new}, \mathbf{D}_{new})$, where there are n_{new} features in the set. The source of $\mathbf{F}_{ref}$ varies depending on the registration strategy, as discussed in Section 6.1.6.

We compute a transformation between volumes by first finding a rough correspondence between the features in $\mathbf{F}_{ref}$ and those in $\mathbf{F}_{new}$ using a symmetric matching algorithm. This rough correspondence will contain true matches (inliers) and false matches (outliers), and so we subsequently employ a model fitting algorithm (RANSAC) to remove outliers and to better estimate the rigid transformation between volumes.

Symmetric Matching

Symmetric matches between feature sets $\mathbf{F}_{ref}$ and $\mathbf{F}_{new}$ are found by first computing the pairwise distances between descriptor vectors $\mathbf{D}_{ref}$ and $\mathbf{D}_{new}$, where distances are computed as the Euclidean norm of the vector differences. If we index vectors $\mathbf{D}_{ref}$ with α, where $\alpha \in \{1, ..., n_{ref}\}$, and index vectors $\mathbf{D}_{new}$ with β, where $\beta \in \{1, ..., n_{new}\}$, then a symmetric match is said to occur when $A = \underset{\alpha}{\operatorname{argmin}} \|\mathbf{D}_{\alpha,ref} - \mathbf{D}_{B,new}\|$ and $B = \underset{\beta}{\operatorname{argmin}} \|\mathbf{D}_{A,ref} - \mathbf{D}_{\beta,new}\|$. The $\mathbf{X}_{ref}$ and $\mathbf{X}_{new}$ corresponding to the symmetric matches are stored as $\mathbf{M}_{ref}$ and $\mathbf{M}_{new}$, where the coordinate positions of the features from symmetric match i are $\mathbf{M}_{i,ref}$ and $\mathbf{M}_{i,new}$ and $i \in \{1, ..., n_{sym}\}$.

RANSAC

While symmetric matching generates a rough correspondence between features in two feature sets, outliers (false matches) prevent computing an accurate registration strictly from the symmetric matches. For this reason, we use a RANSAC (RANdom SAmple and Consensus) algorithm [20, 34] to remove outliers so that a more accurate registration can be computed. In each RANSAC trial, three unique symmetric matches are used with the least-squares registration algorithm in [3] to estimate the transformation, $\mathbf{T}_t$, that maps $\mathbf{M}_{new}$ to $\mathbf{M}_{ref}$, where $t \in \{1, ..., n_{trials}\}$ and $n_{trials} = 10n_{sym}$. The coordinates $\mathbf{M}_{new}$ are then transformed according to $\mathbf{T}_t$, and a support for the trial, S_t, found as the number of matches where $\|\mathbf{M}_{i,ref} - \mathbf{T}_t\mathbf{M}_{i,new}\| \leq d_{ransac}$, where d_{ransac} is a distance threshold that determines the cut-off for when a transformed symmetric match is considered an inlier versus an outlier. A study to determine an appropriate value for d_{ransac} is described in Section 6.2.1. The symmetric matches that make up the support from the trial with the largest S_t are then used again with the least-squares registration algorithm in [3] to estimate the final transformation, $\mathbf{T}_{final}$, that transforms the new volume to the coordinate space of the reference volume.

6.1.5 Interpolation and Mosaicing

Once the transformation, $\mathbf{T}_{final}$, relating the position of $\mathbf{V}_{new}$ to $\mathbf{V}_{ref}$ is determined, we reconstruct $\mathbf{V}_{new}$ in the coordinate system of $\mathbf{V}_{ref}$. We interpolate the transformed volume using a tri-linear interpolation method. Using an eroded mask of the conical data volume, we avoid interpolating the data near the intersection (within a two voxel radius) of the conical 3DUS image and the surrounding padding of zeros, thereby avoiding the blurring effect that would otherwise be seen in the interpolated image. For the application of constructing a large FOV 3DUS mosaic, this newly interpolated image, $\mathbf{V}_{interp}$, is "added" to the mosaic, $\mathbf{V}_{mosaic}$, by use of an averaging method for compounding the existing mosaic and the new volume. Voxels with a value of zero do not factor into the compounding, and therefore do not affect the mosaic.

As the size of $\mathbf{V}_{mosaic}$ is generally larger than $\mathbf{V}_{new}$, we use a windowing operation to limit the region within $\mathbf{V}_{interp}$ for which the interpolation is computed to the subvolume at which the transformed volume resides. Updating a mosaic and computing a compounded image is performed using the same windowing approach. To compound using the averaging method, we maintain two volumes, $\mathbf{\Sigma}_{data}$ and $\mathbf{\Sigma}_{count}$, where $\mathbf{\Sigma}_{data}$ is the summation of interpolated intensities from each volume added to the mosaic, and $\mathbf{\Sigma}_{count}$ the summation of the number of voxels that have contributed to the data in $\mathbf{\Sigma}_{data}$. Therefore, a compounded mosaic image using the averaging method is found as $\mathbf{V}_{mosaic} = \mathbf{\Sigma}_{data}/\mathbf{\Sigma}_{count}$.

6.1.6 Registration Strategies

When registering volumes in a 3DUS sequence, there are two registration strategies that could be employed. One strategy would be to register the next volume in a sequence to the most recently registered volume, a method which will be referred to as the Register-to-Previous (RTP) method. While this method would be suitable for short 3DUS sequences, in the case where long sequences are acquired, this method would accumulate substantial error. An alternate strategy to registering new volumes is to instead use some combination of the features from all volumes in the sequence previous to the current volume, a method which will be referred to as the Register-to-Global (RTG) method.

A naive approach for the RTG method might be to simply retain the features from every volume and store their positions and descriptors in a large database for use in registering subsequent volumes. However, this approach is inefficient and can lead to the storage of repeated and useless information. Rather, we record only the "good" features (i.e. features that are likely to be found in multiple volumes) and maintain a manageable database size by limiting repeated features, therefore allowing for real-time execution of the registration algorithm. To accomplish this, we start by only recording those features that make up the support from a previous 3DUS volume registration. As these corresponding features were found in at least two previous volumes, it is more likely they will be found in subsequent volumes. Also, because features are required to be local minima of the LoG, it makes sense to allow only one feature to occupy a small neighborhood. We keep track of the feature positions for features added to the database by maintaining a volume, $\mathbf{V}_{DB}$, that is the same size as $\mathbf{V}_{mosaic}$ and that indicates the positions of the previously recorded features. A new feature is added to the database only if a feature has not already been recorded at its location. When a new, unique feature is found, its index location in the database is then written to the 27-voxel neighborhood centered at the feature position in $\mathbf{V}_{DB}$. Additionally, the feature position and descriptor vector are recorded to the database.

When registering new volumes, it is unnecessary to use the entire database of features to compare to the features of the new volume, as there will be a large number of features whose position will be nowhere near the volume to register. Rather, the feature set that we use as the reference feature set, $\mathbf{F}_{ref}$, when registering a new volume is a combination of a small subset of the feature database and the features from the previously registered volume. The features that are used from the database are those that resided at the location of the conical data volume in the transformed previous volume, $\mathbf{V}_{interp}$. This is based on the assumption that there are small displacements between consecutive frames. These features can be quickly found by looking at the intersection of $\mathbf{V}_{interp}$ and $\mathbf{V}_{DB}$. Additionally, all features from the previous volume are used because neighboring volumes will have a large number of corresponding features, and these features do not necessarily already exist in the database.

6.2 Validation and Performance

We performed several studies to characterize the behavior of the algorithm. These studies either explored the parameter space of the algorithm to determine suitable ranges for parameter values or explored the algorithm performance for different applications. Data was acquired for these studies using a 3DUS machine (iE33 Echocardiography System with transesophageal X7-2t and transthoracic X7-2 probes, Philips Healthcare, Andover, MA, USA) with the capability to stream 3DUS images to an external PC (Dell Alienware Aurora, Intel Core i7 processor @ 2.67GHz, 6GB RAM, NVIDIA GTX260 graphics card). Several studies were done using images that were acquired at known positions (i.e. known translation and rotation offsets). This used the water tank setup shown in Figure 6.3, which features a translation stage (to which we attached the X7-2 ultrasound transducer) and a rotation stage (to which we attached a porcine heart for imaging).

6.2.1 Parameter Selection

In the registration algorithm, there are four parameters that need to be defined that control how the algorithm operates and performs. These parameters are the feature

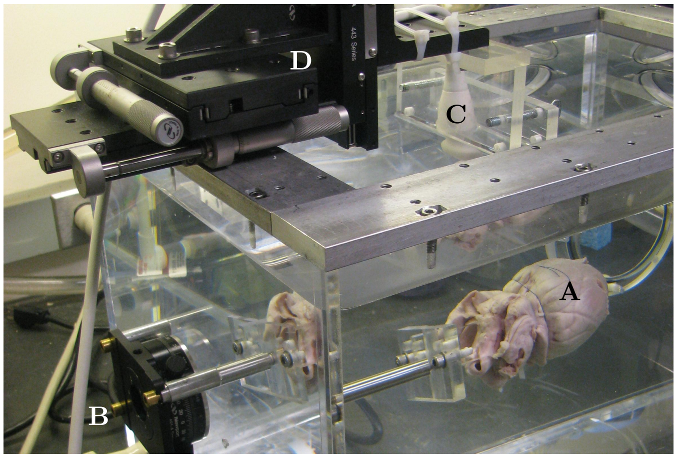


Figure 6.3: Water tank setup that allows for the acquisition of an ultrasound image with a known ultrasound probe position/orientation relative to the object being imaged. Shown are a porcine heart (A) attached to a rotation stage (B), and an ultrasound probe (C) attached to a translation stage (D).

descriptor sample offset scale factor (M_d), RANSAC distance threshold (d_{ransac}), feature scale (σ_f), and tissue threshold (τ_{tissue}). The sample offset scale factor determines, relative to the feature scale, the extent of the sampling grid used in the feature descriptor formation process. The feature scale is assumed to be tuned by the user depending on the image and anatomy of interest, and so it follows that this scale factor should be a constant. An optimal range and value for the scale factor is determined in the first study. Similarly, the RANSAC distance threshold should also not need to be tuned by the user, as it should be made as small as possible so that accurate registrations result, but not so small as to be on the order of the original 3DUS resolution. We show this to be the case in the second study. The feature scale and tissue threshold are dependent on the imaging and anatomy being imaged, as the scale of the dominant features and the 3DUS intensity at prospective feature locations will change for different tissue and anatomic structures. These values are therefore tuned by the user. The effect of varying these values is studied and described in the third and fourth study.

The studies described in this section were performed using images acquired in the water tank (Figure 6.3) using combinations of known translations (0, 3, 6, and 9 mm) and rotations (0, 4, and 8 degrees). The objective of the registration algorithm is to register images at a frame rate of roughly 30Hz, and so this range translates to probe motions upwards to 0.27 meters per second and 242 degrees per second, which is expected to be an upper bound on expected probe motion. The ultrasound volume size was $144 \times 80 \times 112$ voxels with a resolution of $0.75 \times 0.65 \times 0.83$ mm/voxel. For the following studies, the images were registered to an image at 0 mm of translation and 0 degrees of rotation using the described parameters, and registration errors computed according the offset of each image.

Sample Offset Scale Factor, M_d

The size of a feature is related to the designated feature scale at which the LoG is computed. When computing the LoG, for a detection error of less than 0.1%, the width of the LoG kernel is suggested to be roughly 8.5 times the feature scale [31]. Therefore, the extent of the grid used to compute the feature descriptor should be *at least* as large, meaning the corresponding sample offset scale factor would have a value of at least 2.125. If M_d was made to be too large, sample points would be so spread out that the descriptor would include distant image data unrelated to the feature.

We therefore explored how varying M_d would affect registration performance to determine an optimal range for the value of M_d. We did this by registering the images of known position using varying M_d from 1 to 8. The registrations were computed using $d_{ransac} = 1.5$mm, $\tau_{tissue} = 150$, and three different feature scales ($\sigma_f = 1$, 1.5, and 2mm). The three spatial scales were determined qualitatively to be a reasonable range for the feature scale for the given images. As the RANSAC algorithm is a random process, the

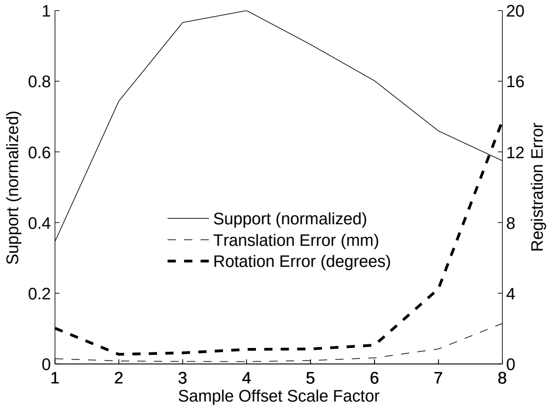


Figure 6.4: Effect of the feature descriptor sample offset scale factor (M_d) on registration performance.

registration results can vary slightly from one execution to the next, and so we performed the registrations 10 times for each feature scale, resulting in 30 trials for each scale factor. The results of this analysis are averaged over all feature scales and trials, a plot of which can be seen in Figure 6.4.

The plot indicates that when M_d is between 3 and 4, the support is at its maximum. Meanwhile, the registration error (both translation and rotation) is roughly constant for M_d between 2 and 6. Therefore, in the interests of generating the maximum support and limiting error, we use $M_d = 3.5$ for the remaining studies.

RANSAC Distance Threshold, d_{ransac}

The RANSAC distance threshold should be made as small as possible to limit registration error, but the lower bound on the threshold value is near the spatial resolution of the 3DUS volume. To show this, we analyzed the effect of the RANSAC distance threshold, d_{ransac}, on the algorithm performance in much the same way as was done for M_d, with the difference being that for this study we fixed M_d at 3.5 while d_{ransac} varied between 0.5 and 5mm. The results were averaged over all feature scales and trials (Figure 6.5). As

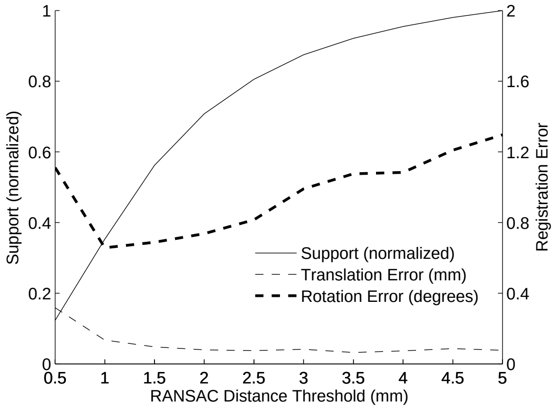
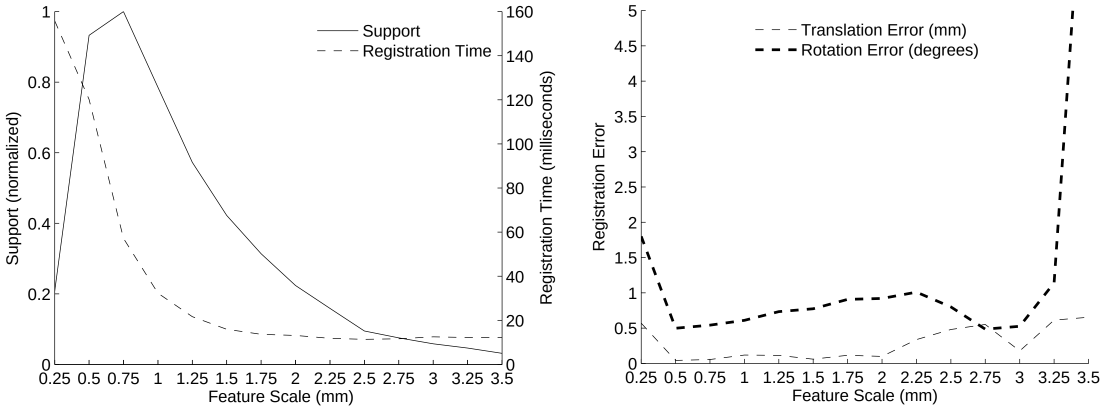


Figure 6.5: Effect of the RANSAC distance threshold (d_{ransac}) on the registration performance.

anticipated, for d_{ransac} below the spatial resolution of the volume (about 0.75mm/voxel) and for large values, the registration performance diminished. Appropriate values for d_{ransac} would then be those between 1 and 2mm, as registration error is comparatively small and constant in this region. Accordingly, we use a d_{ransac} value of 1.5mm for the remaining studies.

Feature Scale, σ_f

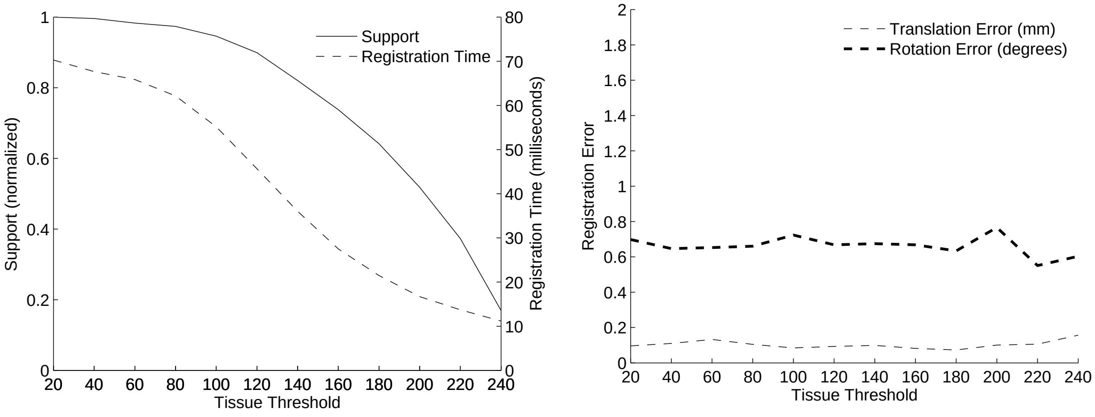
As previously mentioned, an appropriate feature scale at which to detect features will depend on the imaged tissue and anatomy. To elucidate the effect of varying feature scale on registration performance, we computed the registrations for the same images as used in the previous two studies, using the optimal values of $M_d = 3.5$ and $d_{ransac} = 1.5$mm and making $\tau_{tissue} = 150$. The value of the feature scale was varied between 0.25mm and 3.5mm, and similar to the previous studies, several trials (25 trials) were performed for each feature scale. The performance values were then averaged over all trials. The effect of the feature scale on the number of matches found and the registration time can be seen in Figure 6.6(a), while Figure 6.6(b) shows the effect of the scale on the registration accuracy.



(a) (b)

Figure 6.6: Effect of feature scale (σ_f) on (a) the registration support and execution time and (b) the registration accuracy.

With regards to the accuracy, it can be seen in Figure 6.6(b) that in an effort to keep the translation error as low as possible, a value of σ_f between 0.5mm and 2mm should be used. In this range, the rotation error is smallest for smaller σ_f, and so it might be assumed that a σ_f of 0.5mm should be used. However, as seen in Figure 6.6(a), the registration time for small feature scales becomes large due to the increased number of features being detected. It would therefore be prudent to choose a small enough feature scale to maintain registration accuracy, but large enough that the registration time will allow for real-time operation. For example, if the frame rate is 30Hz, the registration time should be $\leq$ 30ms, therefore an appropriate feature scale would be 1-1.5mm. It is important to note that the registration time is dependent on the GPU hardware and architecture. While this study was performed using an NVIDIA GTX260 graphics card, more recent hardware releases would prove to substantially reduce the reported registration times. A more detailed analysis of registration time versus GPU hardware can be found in Section 6.2.5.



(a) (b)

Figure 6.7: Effect of the tissue threshold (τ_{tissue}) on (a) the registration support and execution time and (b) the registration accuracy.

Tissue Threshold, τ_{tissue}

The tissue threshold is primarily used to limit how many features are found in an image, and so it is important to understand its effect on the registration performance. We examine the effects of the tissue threshold on the registration performance using the same water tank images from the previous three studies. For this study, $M_d = 3.5$, $d_{ransac} =$ 1.5mm, and $\sigma_f = 1.0$mm, while we made τ_{tissue} to vary between 20 and 240 (assuming intensities are in the range of 0-255). For each value of τ_{tissue}, 25 trials were performed. The results of the registration performance, shown in Figure 6.7, were then found by averaging across these trials.

For larger values of τ_{tissue}, fewer features are being found in each image and therefore a lower support results. While this has the effect of lowering the registration time, this has little effect on the registration accuracy when τ_{tissue} is below roughly 200. As a result, τ_{tissue} is a suitable parameter to tune to adjust the registration time to below acceptable limits without affecting accuracy. For instance, it can be seen in Figure 6.7 that changing τ_{tissue} from 120 to 170 has almost no effect on the registration accuracy, yet the registration time is cut in half.

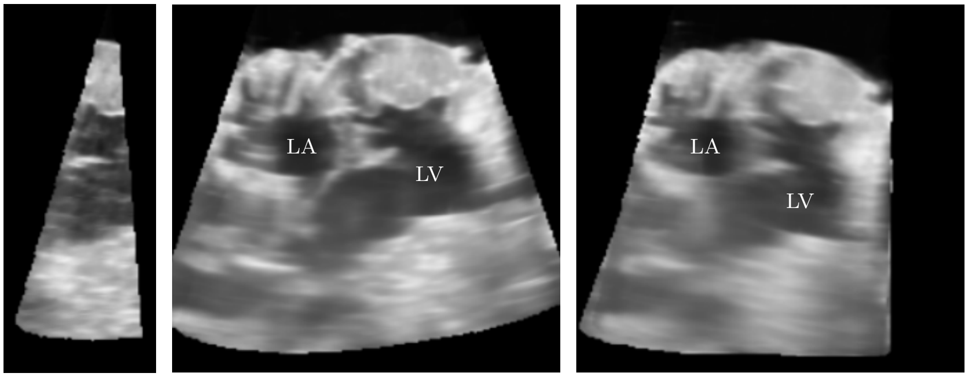


Figure 6.8: Mosaic of a porcine heart in a water tank. (*Left*) Example of the size of one of the volumes that were used to generate the mosaic. (*Center*) Cross-section of the mosaic created using the RTG method. (*Right*) Cross-section of the mosaic created using the RTP method. Notice how the RTP mosaic lacks definition and is a different shape compared to the RTG mosaic, indicating more significant registration error. (LA – left atrium; LV – left ventricle)

6.2.2 Drift Analysis for Different Registration Strategies

As we are proposing to use the presented registration algorithm to generate arbitrarily large 3DUS volumes, it is important that the algorithm accumulate minimal error as more and more volumes are added to the mosaic, especially if the same region is imaged several times throughout an acquisition. To analyze the accumulation of error for the RTP and RTG registration strategies discussed in Section 6.1.6, we conducted a study which analyzed the accumulation of registration error by computing the difference between the first and last frame in a loop trajectory of the ultrasound probe. For this study, we again used the water tank setup shown in Figure 6.3, allowing us to accurately acquire the first and last frames of the loop in the same position. The parameter values used were $M_d = 3.5$, $d_{ransac} = 1.5$mm, $\sigma_f = 1.5$mm, and $\tau_{tissue} = 150$. The trajectories were created using a series of translations and rotations along and about the y-axis of the image, where translations and rotations are traversed in 3mm and 4^o increments, respectively. If we designate a translation as T and rotation as R, then the studied trajectories were as follows:

Table 6.1: Registration strategy comparison – position and orientation error between the first and last images registered in an ultrasound sequence acquired using a loop trajectory of the transducer

		Error	
Trajectory	Registration Strategy	Translation (mm)	Rotation (degrees)
1	RTG	0.36	0.39
	RTP	0.56	1.06
2	RTG	0.28	0.36
	RTP	1.73	2.81
3	RTG	0.27	0.31
	RTP	4.70	4.15
Avg.±	RTG	0.30 ± 0.05	0.35 ± 0.04
Std. Dev.	RTP	2.33 ± 2.13	2.67 ± 1.55

1. T(48mm), T(-48mm)

2. T(48mm), R(20^o), T(-48mm), R(-20^o)

3. T(48mm), R(60^o), T(-48mm), R(-60^o)

The accumulated error for the three trajectories can be seen in Table 6.1. While the RTG method exhibited roughly constant error for the three different trajectories, the RTP method exhibited a larger error than the RTG method for every trajectory, and also accumulated an increasing amount of error as the number of frames and total rotation throughout the trajectory increased. This suggests that the RTG method is better suited for registering several volumes to create a 3DUS mosaic. Qualitative verification of this conclusion can be seen by comparing the mosaics generated using the RTP and RTG methods shown in Figure 6.8 ($M_d = 3.5$, $d_{ransac} = 1.5mm$, $\sigma_f = 1.5$mm, $\tau_{tissue} = 150$), which were created from a left ventricular axis sweep (437 3DUS volumes of size $160 \times 64 \times 208$ voxels and resolution $0.56 \times 0.70 \times 0.58$mm/voxel) of a porcine heart in a water tank. The mosaic created using the RTG method shows clear boundaries and texture within the tissue, while the mosaic created using the RTP method has much less definition and a different

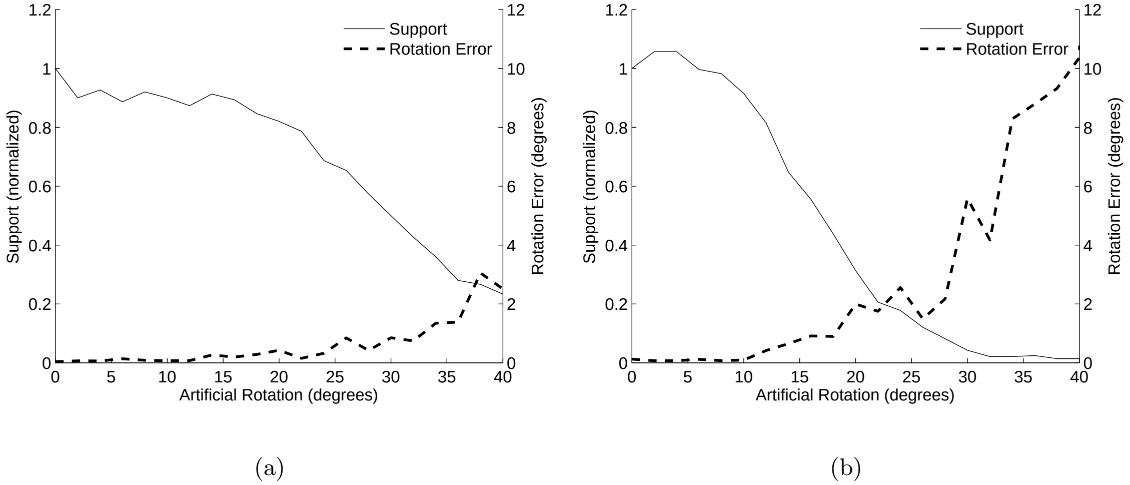


Figure 6.9: Registration accuracy for (a) an artificially rotated clinical 3DUS image and (b) an artificially rotated water tank 3DUS image.

overall shape. This again supports the conclusion that the RTG method is better suited for creating 3DUS mosaics from many images.

6.2.3 Registration Accuracy Under Artificial Rotation

Given that we have chosen to construct rotationally variant feature descriptors, it is helpful to know the extent to which these features can accurately register two images. In an ideal scenario where the imaging is independent of the ultrasound probe orientation to the anatomy, the feature descriptors are affected by rotation but not translation. To analyze the effect of image rotation on the registration accuracy, we perform a study similar to that done in [67], where the stability of feature matching under rotation was analyzed for a registration method that used the rotationally invariant 3D SIFT feature descriptor. In performing a similar study, a rough comparison can be made between the two different feature descriptor formation approaches. For this study, we generated a set of artificially rotated images by applying a known rotational transformation to a baseline image. We then registered the transformed image to the baseline image ($M_d = 3.5$, $d_{ransac} = 1.5$mm,

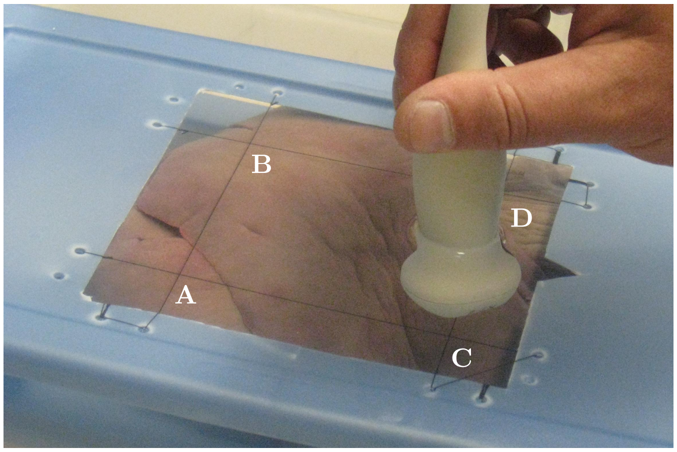


Figure 6.10: Water tank setup showing a porcine liver with surrounding fiducials - suture intersections A, B, C, and D (occluded).

σ_f = 1.5mm, and τ_{tissue} = 150) and computed the registration error. This was done for rotation angles between 0 and 40 degrees. The resulting support (normalized by the support from the 0 degree image registration) and the registration accuracy for both a clinical image (Figure 6.2(a)) and water tank image, both generated using the X7-2 transducer, can be seen in Figure 6.9.

The results are similar to those shown in [67]. As a point of reference, if we compare the angle of rotation at which the normalized support drops below 50%, in the case where the rotationally variant descriptors were used, the rotation angle was roughly 30^o and 15^o for the clinical and water tank images, respectively. In [67], the rotation angle was 15^o for both a clinical and water tank image.

6.2.4 Registration Accuracy in a Water Tank Mosaic

To assess the accuracy of the registration algorithm for the purpose of generating a 3DUS mosaic, we manually selected fiducials in a mosaic and compared the image-based inter-fiducial distances versus the actual inter-fiducial distances. The object being imaged was a porcine liver in a water tank (Figure 6.10), and the fiducials (A, B, C, and D in

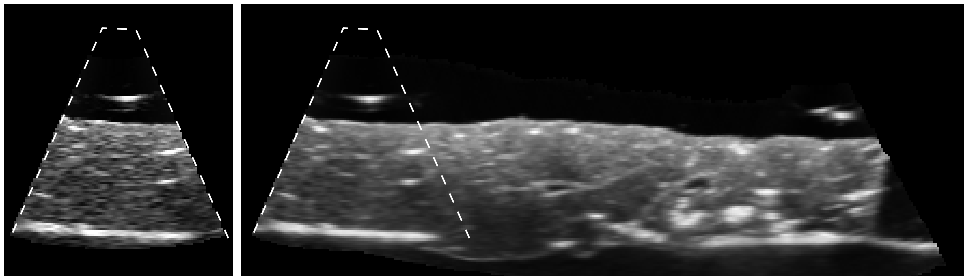


Figure 6.11: Cross-sections from (*left*) a 3DUS volume used to create a mosaic of the porcine liver and (*right*) a 3DUS mosaic created from the sweeping acquisition of a porcine liver. The dotted lines in the image on the right correspond to the size and location of the volume on the left.

Table 6.2: Summary of the mosaic accuracy study which measured inter-fiducial distances AD and BC. Variability is reported as *avg.* ± *std.dev.*

	AD	BC	Both
n	15	15	30
Volumes in Mosaic	68.9 ± 7.9	65.2 ± 9.2	–
Actual Length (mm)	121	118	–
Length in Mosaic (mm)	120.5 ± 0.1	116.7 ± 0.9	–
Error (mm)	0.9 ± 0.8	1.4 ± 0.7	1.1 ± 0.8
Error (%)	0.72 ± 0.64	1.20 ± 0.59	0.96 ± 0.65

Figure 6.10) were the intersections of sutures placed in a grid pattern directly above the liver. We acquired sweeping ultrasound data (using the X7-2 ultrasound transducer) by starting at fiducial A or B and ending at D or C, respectively. Typically 50-80 volumes were acquired in each sweeping acquisition, where each volume had dimensions of $144 \times 112 \times 112$ voxels and a resolution of $0.55 \times 0.54 \times 0.63$mm/voxel. For each fiducial pair (AD or BC), 15 mosaics were created using the RTG strategy and the fiducials in those mosaics manually selected. A cross-section from one of the mosaics is shown in Figure 6.11. The parameters used for the registration were $M_d = 3.5$, $d_{ransac} = 1.5$mm, $\sigma_f = 1.5$mm, and $\tau_{tissue} = 150$. A summary of the results from this study are shown in Table 6.2. With an average error of less than 1%, the presented registration algorithm is comparable to the 3D SIFT registration algorithm in [67], which reported an average error of 2% for a similar study.

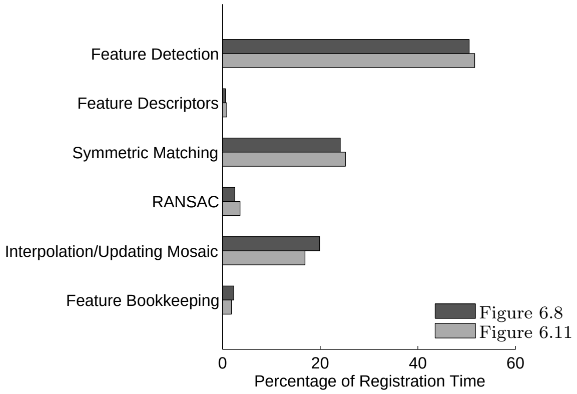


Figure 6.12: Average running times for the components of the registration algorithm relative to the average total registration time. Results are for the mosaics shown in Figures 6.8 (using the RTG method) and 6.11, where the average registration times were 28.7ms and 24.4ms, respectively.

6.2.5 Registration Running Times

The presented feature-based registration algorithm can operate in real-time (less than 30ms), but it is helpful to know how the registration time is distributed over the different components of the algorithm. For this, we have included a detailed time analysis (Figure 6.12) of the registrations performed to compute the mosaics shown in Figures 6.8 and 6.11. For these mosaics, the same M_d and d_{ransac} values were used as found in the studies described in Sections 6.2.1 and 6.2.1, while τ_{tissue} and σ_f were tuned for the different types of images. The time analysis does not include the time to render the mosaic as this can be done in a separate thread.

To determine the effect of the graphics card on the registration time, we also performed the same analysis as was done above but instead of using an NVIDIA GTX 260 (which has 192 cores), we used an NVIDIA GeForce 8600M GT (which has 32 cores). We found nearly identical results with regard to the percentage of time that each component took, however, the average total registration time was roughly six times longer (about 170ms). This indicates a linear relationship between the number of cores and the registration time. This also suggests that if for instance the same registrations were performed on the latest NVIDIA hardware (i.e. NVIDIA GTX 580 with 512 cores), a 2.6× speed increase could be achieved (i.e. registration time decrease from 30ms to 12ms).

6.3 Discussion

6.3.1 Registration Algorithm Performance

In this chapter we have presented a feature-based 3DUS rigid registration algorithm capable of registering volumes in a 3DUS sequence in real-time. The registration algorithm is most closely related to the method in [67], which is a SIFT-based method that ensures that features are detected at multiple spatial scales and also that feature descriptors are scale and rotationally invariant. This is a consequence of the original design of SIFT, which was designed for detecting and matching features in 2D photos [56]. However, while features in 2D photos may appear at different scales simply by changing the position and orientation of the camera relative to the imaged objects, features in 3DUS always appear at the same scale (assuming the voxel resolution is taken into consideration). Also, in registering volumes from a live 3DUS sequence, the inter-frame displacement will be small, and so the need for a feature descriptor that can handle large rotational and translational displacements becomes unnecessary. It has also been shown that ultrasound is not a rotationally invariant imaging modality [29, 83], as the appearance of anatomy is dependent on the direction of acoustic propagation relative to the imaged structures, making rotationally invariant feature descriptors unnecessary.

We therefore only search for features at a single, user-specified spatial scale. We also devised a simple rotationally variant feature descriptor formation method. The simplified feature detection and descriptor formation methods, as well as using a GPU accelerated framework, results in a 2000× speed increase (from 1 minute down to 30ms for single volume registration) compared to the method in [67]. While a substantial speed increased was realized, we showed in several validation studies that this was not at the expense of registration accuracy. On the contrary, similar or better performance was found in those studies (Sec. 6.2.3 and Sec. 6.2.4) where a similar study was performed for the 3D SIFT method.

Comparable registration accuracy and fast execution was made possible through the use of the RTG registration strategy. In maintaining a minimal but representative set of global features and storing and indexing these features efficiently, we were able to limit the accumulation of registration error, even when mosaicing several hundred 3DUS volumes. The most promising results for the RTG method were those described in Sec. 6.2.2, where for loop trajectories, the RTG method accumulated less than 0.5mm and 0.5^o of translation and rotation error, respectively.

The presented registration algorithm operates under a small-displacement assumption to compute accurate transformations in real-time. While it is tempting to define “small” as a percentage of volume size, in the case of the presented feature-based registration method, it is rather a function of image content. The method requires a coincidence of features from frame to frame, and so more important than volume overlap is overlap of salient regions. It is important to take this into consideration when registering anatomy with large “dead zones” that have little or no saliency (such as in the blood pools in the heart), as these regions will not contribute to the registration.

While the algorithm is fast and accurate, a drawback of the presented method is that acceptable ranges for the feature scale and tissue threshold need to be determined so that for an imaging application, registrations can be computed in real-time. While we did use similar parameter values in our studies, these values are not guaranteed to register volumes for every application in real-time. There are, however, options to compensate for this drawback. As we showed in the validation studies, the feature scale and tissue threshold can be tuned to bring down the registration time with minimal effect on the registration accuracy. This suggests that an auto-tuning method for these parameters could be implemented. A preliminary auto-tuner could be used to analyze the first few frames from a sequence to help select appropriate values for the feature scale and tissue threshold to optimize for speed and accuracy. An online auto-tuner could then adjust the tissue threshold (as the feature scale needs to remain constant) to keep the registration time within acceptable limits.

The registration algorithm makes it possible to register 3DUS volumes online without the need for external tracking systems (such as electromagnetic or optical tracking systems), which means that when acquiring data to use with the presented method, any existing 3DUS system should suffice. A limitation of the presented method, however, compared to systems that use a tracking system, is that because a continuous stream of 3DUS volumes is required, any interruption in that stream, such as lifting the probe from the object being imaged, will likely cause subsequent volume registrations to fail under the RTG and RTP registration strategy. For this reason, tracking systems can still be a valuable registration tool, especially if used in combination with an image-based registration method.

6.3.2 Applications of Registration Algorithm

While we have shown the ability of the registration algorithm to mosaic volumes of a porcine heart and liver in a water tank (Figures 6.8 and 6.11), we have also had success creating mosaics of ECG-gated sweeping acquisitions of beating hearts in clinical cases. An example of this is shown Figure 6.14, which was created from a transthoracic subcostal acquisition using the X7-2 transducer. We have also used the registration algorithm to

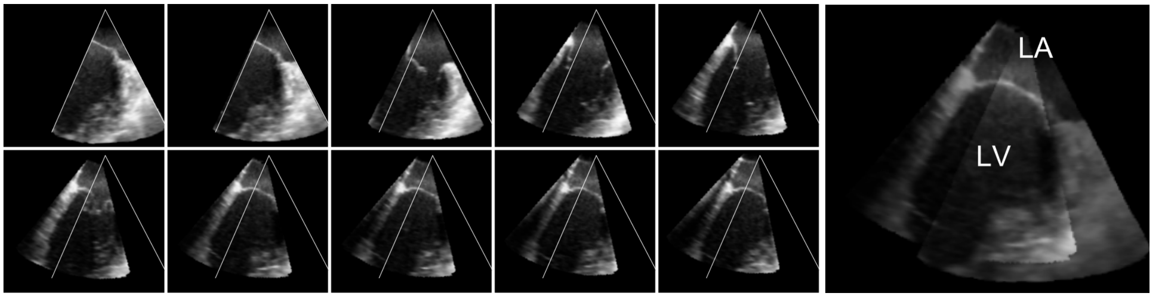


Figure 6.13: (*Small images on left*) Stabilized (registered) images from a live 3DUS sequence of the beating heart (in order from left to right and top to bottom) acquired while rotating the transesophageal probe. The white lines show the location of the first frame relative to the stabilized frame. (*Large image on right*) An overlay of gated images from before and after the probe was rotated, showing that the registration correctly stabilized the sequence. This also shows that the registration algorithm can handle small amounts of nonrigid deformation. (LA – left atrium, LV – left ventricle)

stabilize live clinical 3DUS images of the heart relative to probe movement and respiration, thereby making the objects within the image appear stationary while the image boundary itself moves. An example of this is shown in Figure 6.13, which was created from an acquisition in which X7-2t probe was rotated during a transesophageal acquisition of the mitral valve. The latter of these applications suggests that the algorithm is capable of handling small amounts of nonlinear deformation. The result in Figure 6.15 also shows how registration along with compositing can serve to denoise a clinical 3DUS volume and provide for a clearer differentiation between blood pools and tissue.

6.3.3 Future Work

The presented work has obvious applications other than those shown, such as in hepatic tumor location and intervention, echocardiography and intracardiac echocardiography, and fetal ultrasound. However, to be more robust and useful for these applications, there are several aspects of the algorithm that could be improved. One particular area is accounting for nonlinear deformation. This becomes especially important for applications,

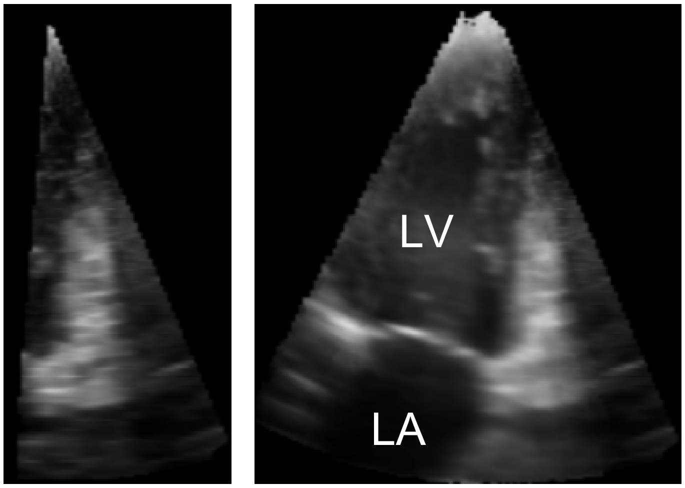


Figure 6.14: (*Left*) Cross-section showing the size of the volumes in an ECG-gated sequence used to create the mosaic on the right. The mosaic was created from 13 3DUS volumes in a sweeping subcostal acquisition of the mitral valve. (LA – left atrium; LV – left ventricle)

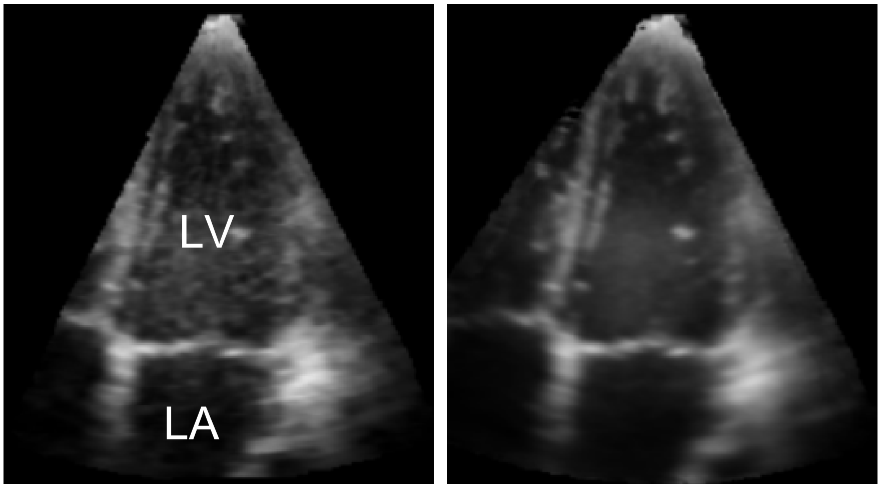


Figure 6.15: (*Left*) Original 3DUS image of the mitral valve acquired using a subcostal approach with the X7-2 probe and (*Right*) the same cross-section after several frames from the same time in the cardiac cycle were registered and composited to the original, showing how an image can be denoised. (LA – left atrium; LV – left ventricle)

such as when imaging the liver, where the probe is in direct contact with the object being imaged and therefore non-rigidly deforms the object, especially near the probe. Other areas of possible improvement are to integrate a real-time temporal registration algorithm based on the ECG signal along with the presented method so that, for instance, a large FOV 4D image of the heart could be constructed in real-time. Lastly, in order to composite images that were acquired in acquisitions where the probe underwent large rotational displacements, it would be best to use instead of averaging a more elegant compositing method (for example, [75]) that takes into account the appearance of anatomy under largely varying probe orientations. The challenges with the proposed future work are not the methods themselves, but integrating the methods into the current system and maintaining a real-time architecture.

Chapter 7

Four-Dimensional Mitral Leaflet Segmentation and Tracking

Gathering detailed information about valve geometry and dynamics on a patient-specific basis is rarely performed. Yet, this becomes increasingly necessary for those studies that want to simulate valve closure for surgical planning purposes [32, 33, 10]. The problem is that a limited number of methods exist that are able to accurately and robustly segment a patient-specific 3D valve and its components with minimal input and interaction required by the user. A volumetric segmentation of the leaflets using an intensity-based level set method was shown in [85], but the segmentation method was not able to isolate the valve from the surrounding tissue and could not separate anterior from posterior leaflets upon coaptation. The work by Burlina *et al.*, 2009 uses a level set method and thin tissue detector to define the valve geometry and surroundings, but requires extensive user interaction and relies on the assumption of a planar annulus [10]. Furthermore, the segmentation is performed in only a single frame showing an open valve. The closed configuration of the valve is estimated by modeling the mechanics of the valve, but the accuracy of this approach is not validated.

The work by Ionasec *et al.*, 2010 described a sophisticated system that fits a mitral and aortic model to the respective valve using machine learning techniques and a large database of manually delineated points [37]. In the context of mitral valve segmentation, however, the mitral valve model lacked fine leaflet detail due to the model being fit to only a few locations. Additionally, the model appeared to only roughly estimate the coaptation line and did not appear to be able to generate an estimate of coaptation length, which is a measure commonly used by clinicians to assess valve competency.

To address these issues, we present a 4D mitral valve segmentation method that can generate a detailed 4D patient-specific annulus and leaflet geometry. As most modeling efforts and clinical interest revolves around the closing of the valve, the method is designed to capture the valve geometry during this phase of the cardiac cycle. However, the method could easily be altered to find the leaflet geometry throughout an entire 4D ultrasound (4DUS) sequence. The method requires only the selection of two frames from an ultrasound sequence – one before the valve starts to close and another after the valve closes – and the selection of a point near the center of the closed valve. The method then automatically finds the annulus and leaflets during valve closure. By taking into account valve behavior and handling leaflet collisions, a detailed coaptation line and surface are generated. It is important to make the distinction that we are not simulating the mechanics of the valve, but rather are presenting a method to delineate the valve as seen in a 4DUS sequence. The extracted geometry is intended for inputs to mechanical models or surgical simulators, improved visualization, diagnostics, or for inverse modeling purposes.

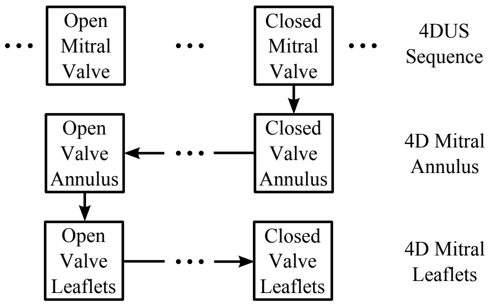


Figure 7.1: Summary of 4DUS segmentation of the mitral valve. A user manually specifies the frames at the start and end of valve closure, and also a point near the center of the closed mitral valve. The annulus is found through segmentation and then tracking, starting from the closed valve and proceeding to the open. The leaflets are then found through segmentation and then tracking, starting from the open valve and proceeding to the closed.

7.1 Methods and Materials

7.1.1 Constructing Geometric Priors

The mitral valve segmentation method operates by first constructing a geometric prior of the valve which is then propagated to all other frames of interest. The valve geometry is constructed by first finding the annulus, which is then used to enforce constraints about where to find the leaflets. To find the annulus, the user-selected frame showing a closed mitral valve and the selected valve center point are used in the 4D annulus segmentation method described in Chapter 3. The annulus in the frame selected by the user as showing the mitral valve in the open configuration before the valve begins to close is then used in the direct leaflet segmentation method described in Chapter 4 to generate a detailed patient-specific triangular mesh at the leaflets. The following sections describe how the mesh is evolved during valve closure such that the geometry mirrors that of the valve in the 4DUS images.

7.1.2 Evolving the Leaflet Mesh

The mitral valve mesh is evolved frame to frame by treating the mesh as an active surface [92, 113], where the nodes of the mesh at the annulus are fixed according to the annulus segmentation and tracking results, and the rest of the mesh is free to evolve. The active surface is encouraged to track the top (atrial) surface of the leaflets as seen in the 3DUS images, regulated using internal edge and bending forces, and designed to handle collisions between leaflets. The active surface is then evolved according to the force equation

$$\mathbf{F}_{valve} = w_{image}\mathbf{F}_{image} + w_{edge}\mathbf{F}_{edge} + w_{bend}\mathbf{F}_{bend} + w_{collision}\mathbf{F}_{collision} \tag{7.1}$$

where w_{image}, w_{edge}, w_{bend}, and $w_{collision}$ are scalar weights. The nodes, $\mathbf{x}$, of the mesh (except for annulus nodes) are then evolved such that $\mathbf{x}^{k+1} = \mathbf{x}^{k} + dt\mathbf{F}_{valve}$, where dt is a time step. The annulus nodes, $\mathbf{x}_{annulus}$, are evolved according to the displacement, $\mathbf{D}_{annulus}$, and correspondence determined from tracking the annulus from frame to frame. Displacing the annulus nodes is carried out over $k_{annulus}$ time steps, such that $\mathbf{x}^{k+1}_{annulus} = \mathbf{x}^{k}_{annulus} + \delta_{annulus}\left(\frac{1}{k_{annulus}}\right)\mathbf{D}_{annulus}$, where $\delta_{annulus} = 1$ when $k < k_{annulus}$ and zero otherwise.

Image Force

The image force drives the mesh so that it tracks the top surface of the leaflets as seen in the 3DUS volumes. One component of the force is derived from the image gradient of a Gaussian convolved thin tissue detector image [83], $\nabla \mathrm{G}_{\sigma}\left(\mathbf{TTD}\right)$. This component drives the valve mesh toward the center of the leaflets and helps ensure that leaflets are driven to roughly the desired location even in the presence of large displacements. We encourage the mesh to find the atrial surface of the leaflets using the gradient of the target frame's ultrasound intensity, $\mathbf{I}_{target}$, along the node normal, $\mathbf{N}$. Having already defined the normal

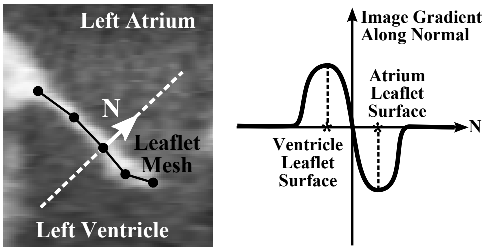


Figure 7.2: Simplified depiction of the image gradient along the node normal. (*Left*) Ultrasound volume shown with the position of a mesh and the direction of a node normal. (*Right*) A simplified graph showing the gradient of ultrasound intensity along the normal. To capture the atrial side of the leaflets, nodes are forced in a direction so as to minimize the value of the gradient.

at each node to point toward the atrial side of the leaflets, the atrial surface of the leaflets should be at a location where $\frac{\partial \mathbf{I}_{target}}{\partial \mathbf{N}}$ is at a minimum (Figure 7.2). The image force is then specified as

$$\mathbf{F}_{image} = \nabla \mathrm{G}_{\sigma} \left(\mathbf{TTD}\right) \cdot \mathbf{N} - \delta_{image} \frac{\partial^2 \mathbf{I}_{target}}{\partial \mathbf{N}^2} \tag{7.2}$$

where $\delta_{image} = 1$ when a node is at a location where $\frac{\partial \mathbf{I}_{target}}{\partial \mathbf{N}} < 0$ and zero otherwise. This is to prevent a node that is below the leaflet from being pushed even further below the leaflets.

Edge Force and Artificial Strain

We maintain the geometric prior as determined from the initial leaflet segmentation by regulating the edge lengths of the mesh. We do this by imparting the edge force

$$\mathbf{F}_{edge} = \sum_{e=1}^{E} \delta_{edge} \left(L_e - L_{e,o}\right) \tag{7.3}$$

where $\delta_{edge} = \delta_c$ when $L_e - L_{e,o} < 0$ (i.e. edge is in compression) and $\delta_{edge} = \delta_t$ when $L_e - L_{e,o} > 0$ (i.e. edge is in tension). The original edge lengths are represented by $L_{e,o}$ and the number of edges attached to each node by E. For our design, we make $\delta_c < \delta_t$ as a way of penalizing longer edges over shorter.

A key issue in creating segmented models of the mitral valve is identifying the regions where the leaflets overlap when closed, as this determines the propensity for the valve to leak. This is challenging because it is impossible to discern separate leaflets in this coaptation region once the valve is closed. This necessitates tracking the leaflets as the valve closes, and approximating the stretch that ensues as the leaflets are pressurized. Studies on the mechanical properties of the valve have shown that leaflets stretch by roughly 35% in the radial direction and 20% in the circumferential direction [28] due to the pressure gradient across the valve, and because the leaflet is constrained at the annulus and by chords. As accurate chordal structure is extremely difficult to obtain from 3DUS, leaflet strain is applied artificially throughout the closure of the valve. Having determined the radial and circumferential components of each edge when the leaflet mesh is first segmented, the reference edge lengths, $L_{e,o}$ are adjusted accordingly throughout the valve. Making $\delta_c < \delta_t$ makes this artificial strain not a hard constraint but a soft constraint, and making δ_t large ensures that the leaflets do not stretch much more than the applied artificial strain. In our design, the applied strain rate is such that the leaflets are fully strained by halfway through leaflet closure to ensure that leaflets have the opportunity to coapt.

Bending Force

Due to the noise in 3DUS images, the image force has the potential to displace the leaflet mesh in ways uncharacteristic of an actual valve and in ways which cannot be corrected by the edge force. We therefore impart a bending force which tends to straighten the leaflets to further regularize the surface geometry.

$$\mathbf{F}_{bend} = \arccos\left(\mathbf{N}_i \cdot \mathbf{N}_j\right) \tag{7.4}$$

The bending force is computed at each edge which is shared by two triangular faces whose respective normals are $\mathbf{N}_i$ and $\mathbf{N}_j$. The force is then imparted appropriately to the edge nodes and the nodes opposite the edge from each face.

Collision Force

To resolve collisions between leaflets and ultimately allow for the formation of a detailed coaptation surface, a collision force is computed between edges that are close to colliding, where potentially colliding edges are determined from a grid-based culling operation. If the distance between two edges is d_c, the force between the edges is $f_c = \exp(d_c)$. The collision force at each node is then found as

$$\mathbf{F}_{collision} = \sum_{e=1}^{E} f_{c,e} \tag{7.5}$$

where E again represents the number of edges attached to a node.

Tether Force

As previously mentioned, segmenting the chordal structure of the mitral valve from 3DUS is extremely difficult. The chords act to constrain the free edge of the leaflets. Without chords, the free edge of the leaflets has the potential to flail uncontrollably. The same is true for the leaflet mesh in the presence of the image force. We therefore found it necessary to add a comparatively small tethering force to the free edge of the leaflets. This is done by redirecting the image force at the free edge by adding a component to the normal that points toward the left ventricle. If the magnitude of the image force is $f_{image,o}$, which is directed along node normal $\mathbf{N}$, then the new image force at the free edge is

$$\mathbf{F}_{image} = \left(\frac{f_{image,o}}{2}\right) \frac{\mathbf{N} + \mathbf{A}_{valve}}{|\mathbf{N} + \mathbf{A}_{valve}|} \tag{7.6}$$

where $\mathbf{A}_{valve}$ is a vector pointing along the axis of the valve toward the left ventricle which is determined when the leaflet mesh is first segmented.

7.2 Results

We qualitatively assessed the usefulness of the presented method for 4D mitral valve segmentation using data acquired using a transesophageal approach (iE33 Echocardiography System with a transthoracic X7-2t probe, Philips Healthcare, Andover, MA, USA). The data was reconstructed into a high temporal resolution sequence using the method described in Chapter 5. Frames at the start and end of mitral valve closure and a point near the center of the closed mitral valve were manually selected. The valve weights ($w_{image} = 0.05, w_{bend} = 0.15, w_{edge} = 0.02, w_{collision} = 2, \delta_t = 5, \delta_c = 1, dt = 1$) were tuned on a single data set and the method validated on two others.

The results of the segmentation for the two data sets – one showing a normal mitral valve and another showing a stenotic mitral valve – can be seen in Figure 7.3. In showing the mesh location relative to the ultrasound images, it can be seen that the segmentation accurately finds the valve location throughout valve closure. Additionally, the 3D mesh shows that a detailed coaptation surface is delineated. The same parameters were used for both studies, with the exception that the artificial strains for the stenotic mitral valve were made to be half those of the normal (i.e. instead of using 35% strain in the radial direction and 20% in the circumferential, we used 17% and 10%, respectively), which is an estimate for the mechanical properties of stenotic leaflet tissue.

7.3 Discussion

The presented mitral valve segmentation method is capable of finding the location of the mitral valve throughout valve closure, and could be modified to segment an entire cardiac cycle. The benefit of the presented method is that, with minimal user input and no user interaction, the method can segment the mitral valve in each frame during valve

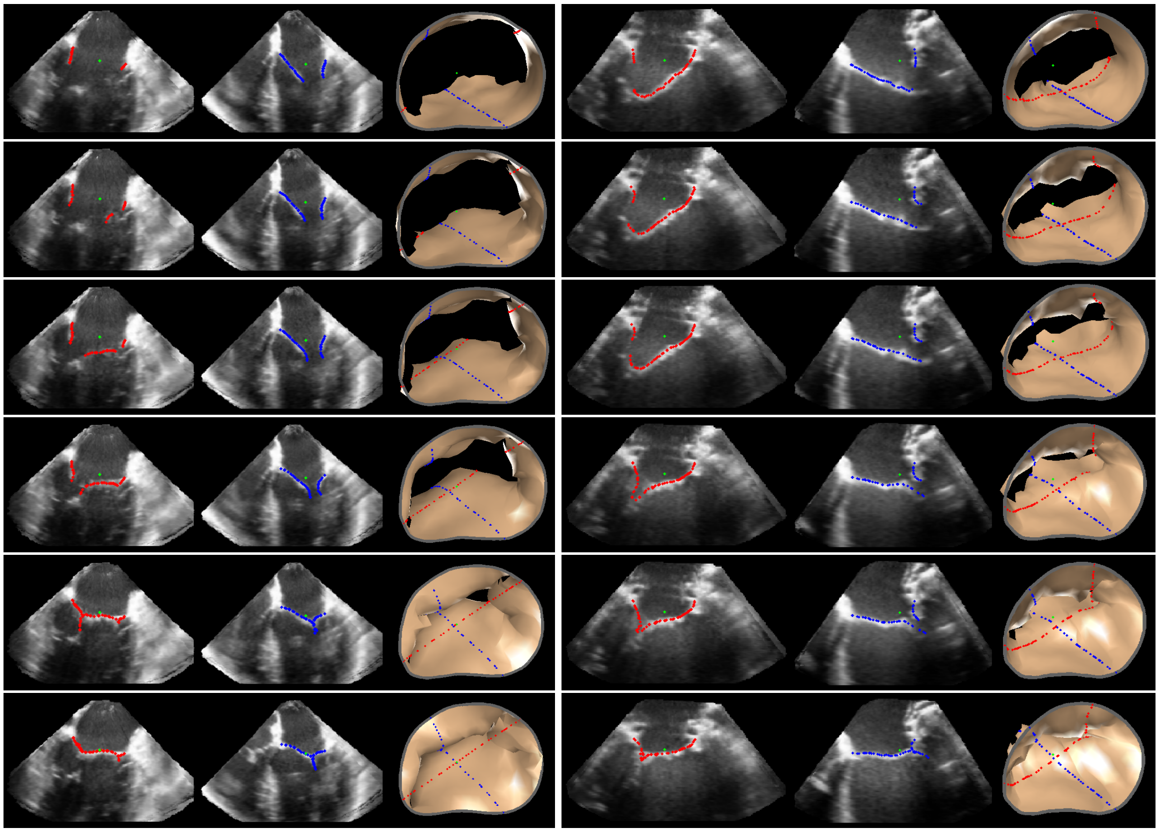


Figure 7.3: Results of the mitral valve segmentation method during valve closure for (*1^{st} Column*) a normal mitral valve and (*2^{nd} Column*) a stenotic mitral valve. The comparison of the mesh location in the image planes (red and blue points) relative to the leaflets as seen in the images show the effectiveness of the 4D mitral valve segmentation method for ultrasound.

closure and is capable of generating a detailed, patient specific coaptation region. This is the first method capable of segmenting such patient-specific detail from a 4DUS sequence. The method benefits from assuming that a geometric prior can be very simply altered and displaced to track the desired anatomy throughout a give sequence. Were we not to assume a geometric prior and rather just segment the leaflets as seen in each separate image, the approach would be similar to that in [37], where a detailed coaptation region was not found.

As pointed out previously, the presented method is not performing mechanical modeling of the mitral valve. However, the method could be used in several aspects of mechanical modeling. The construction of the geometric priors could be used as a modeling input. Material properties could be determined by finding the strain values in the radial and circumferential directions that results in the best segmentation. Lastly, the 4D segmentations can be used to validate their predictions. As such, coupling of segmentation and mechanical modeling could prove to be beneficial.

The constructed geometric priors in this study did not include the mitral valve chords. These structures are too small to be accurately segmented from three-dimensional ultrasound. For this reason, several assumptions, such as artificial strain and tethering forces, had to be made in the segmentation method to account for this lack of information. Future efforts will include specializing ultrasound data acquisition for the collection of the desired chords.

Chapter 8

Conclusions and Future Work

This work presented an image processing pipeline that allows, compared to existing methods, for the most accurate and highest temporal resolution imaging of the heart and heart valves using ultrasound. As the heart has several fast moving valves and structures, the ability to quickly generate ultrasound sequences at millisecond resolution using standard clinical imaging protocols promises to be a valuable clinical tool for understanding cardiac behavior and improving visualization, diagnostics, and surgical planning.

This work also presented a valve segmentation framework whereby the valves of the heart can be robustly segmented from ultrasound images even in the presence of noise and anatomic variation. This was shown for the specific case of mitral valve segmentation. Being able to visualize the progression of mitral valve closure using high temporal resolution imaging, we were able to generate the most accurate and detailed mitral valve segmentation as compared to previous methods. This included, for the first time, an accurate segmentation of the coaptation region. This represents the first time that the annulus and leaflets suitable for mechanical models can be generated from ultrasound data with minimal user-input. Additionally, the detailed, patient-specific, four-dimensional nature of the segmentations provide a means to validate mechanical models using clinical ultrasound data.

8.1 Valve Segmentation

This work presented methods by which a detailed patient-specific mitral valve model can be found from four-dimensional ultrasound data. The methods are designed as an improvement to currently existing methods which either require extensive user-interaction [10] or produce geometries that lack patient-specific detail [37]. The methods are also an alternative to systems that require extensive overheard, specifically in the form of large databases of manually delineated points for use in machine learning [37].

The mitral valve segmentation method operates by first using prior knowledge about the valve geometry to find an annulus. This is done by delineating the annulus in an ultrasound frame with a closed valve by assuming its location to be the point where the thin leaflet tissue meets the thicker surrounding tissue of the heart. Knowledge about the valve shape and motion are then used to track the annulus throughout a cardiac cycle using both a valve state predictor and constrained optical flow algorithm. The annulus is then used throughout the sequence as a means of limiting the search for the leaflets. Knowing that the leaflets are attached to the annulus and rotate about its location, the search for the leaflets is restricted to the immediately surrounding areas, and the geometry of the leaflets constructed such that it conforms to known variations (i.e. leaflets do not exhibit discontinuities or extremely sharp curvature). Again using knowledge about the shape and motion of the valve, along with general knowledge on the mechanical properties of leaflet tissue, the leaflet geometry is tracked to the remaining frames, using the pre-segmented annulus as a known boundary condition and constraint. In progressively building up the valve, starting from prior knowledge to construct an annulus and then using the annulus to construct the leaflets, an accurate and robust mitral valve model is constructed with minimal user-input and no intermediate user-interaction.

Complete Automation

Because the inputs to the mitral valve segmentation method are easily defined, and because the methods are fairly insensitive to the inputs as shown in the validation studies in Chapters 2 and 3, it should be possible to automate the valve segmentation method so that no user-input is required. For instance, for defining the mitral valve center point, it stands to reason that a method could be used such as that described by Ionasec *et al.*, 2010 [37], which automatically found valve feature points using a database of manually delineated points and machine learning techniques. There are, however, two problems with automating the segmentation method using this approach. First, this adds overhead to a system because to make a functional system, data first needs to be collected and manually delineated. This makes the system less accessible to other researchers in the medical image processing community. Secondly, the robustness of the method is unknown due to the variation in patient-specific anatomy, disease states, and imaging protocols. The only way this method could be made robust is by including hundreds, if not thousands, of manually delineated images from all shapes and sizes of valves with varying disease states imaged under varying imaging protocols to show both what is the valve center point and what is *not* the valve center point.

Another method that might serve well to automate the mitral valve segmentation method by picking the valve center point is the method presented by Van Stralen *et al.*, 2008 [97], which was a method by which the left ventricular long axis and the mitral valve plane were automatically found in a 3DUS sequence. The intersection of the defined axis and plane could be used to define the center point. The only foreseeable problem with this method is that it appears that a larger portion of the left ventricular volume needs to be visible to be able to automatically delineate the long axis. This becomes problematic for zoomed images of the mitral valve where a limited amount of the left ventricle is visible.

Assuming the valve center point can be automatically found, the only other inputs to the algorithm are the selection of two frames – one just before and after valve closure. As we have access to the ECG signal, which can be used to determine the temporal location of a frame relative to the cardiac cycle, it could be proposed that the desired frames be selected based on their known positions relative to the QRS complex in the ECG signal. However, as we have studied mostly diseased valves, we are hesitant to correlate valve position and orientation based on frame location relative to the QRS complex, as valve behavior likely varies due to disease states. However, a more appealing approach might be to use an image-based pseudo-ECG signal, such as those shown in [9, 42]. Using the images to determine temporal location in the cardiac cycle would be a much more robust means of finding frames just before and after valve closure, and would hopefully be able to account for variations in valve motion due to disease state.

Mitral Chord Segmentation

The focus of this work was to describe methods by which standard clinical images could be used to generate as much of the mitral valve geometry as necessary for mitral valve closure simulation. Unfortunately, using the current imaging standards and the latest 3DUS technology, segmenting accurate chordal structure from 3DUS remains a challenge. There are, however, non-standard clinical practices that might allow for chordal segmentation and thereby allow for a more complete mitral valve model to be constructed from ultrasound data. One possible approach is to image the chordal structure from several different positions and angles using a small field-of-view (which should serve to improve spatial resolution), and to then mosaic the images into one larger image possibly using the method described in Chapter 6. Because the chords would be imaged from varying positions and angle, a more accurate structure would hopefully be able to be found, especially if denoising is possible from the repetition of information from the same region.

An alternative approach would be to use a high-frequency 2D ultrasound probe (which would generate images with much higher spatial resolution) and progressively scan the left ventricular long axis to image the chords. The images could then be assembled using a freehand 3DUS technique (which generally requires some external tracking system) to compile the 2D images into a 3D volume showing the mitral valve chords [70, 111, 88]. The problem with this approach is that additional hardware in the form of an external tracking system (i.e. mechanical, electromagnetic, or optical) would be required, which would limit its use in the clinical environment.

Mitral Valve Modeling and Surgical Simulation

Because of the complexity of the mitral valve and the challenges associated with mitral valve surgery, several efforts are underway to simulate and predict mitral valve closure. Recent efforts used either generic models that lacked patient-specific detail [17, 52, 47] or generated the models using invasive approaches [46, 52, 32, 33]. The work presented in this thesis describes methods by which these models can be generated (at least in part) using standard clinical data. Additionally, the 4D annulus and leaflet segmentations allow for the validation of these models on a patient-specific basis.

While chords are not currently included in our valve segmentations, the combination of our segmentation methods with a modeling approach could be used to determine how sensitive the valve simulations are to the chordal structure. The necessary accuracy of chordal structure segmentation could then be assessed. Additionally, an inverse modeling approach could be taken in that if the annulus and leaflet geometry are known throughout valve closure, there may be ways to estimate where chords *should be* based on contradictions of the modeling with the 4D segmentations. Knowing probable chord locations will then allow for simulated surgical repair and subsequent surgical planning.

Segmenting Other Valves in the Heart

While we have shown the effectiveness of our methods in mitral valve segmentation, it is reasonable to assume that the methods could be applied to other valves in the heart. The tricuspid valve, being the most similar to the mitral valve in that it has a similarly shaped annulus and has a chordal structure, is the most likely candidate. The pulmonary and aortic valves, however, have a distinctly different shape than the mitral in that they are tri-leaflet valves with three large commissural regions. Nonetheless, they operate on much the same principles on which the mitral valve segmentations were built, that the annulus can be located by finding where the closed leaflets intersect the surrounding wall, and that the leaflets rotate about the annulus. The advantage of our methods for pulmonary and aortic valve segmentation is that, because these valves do not have chords, no additional segmentation would be needed. Something that would have to be accounted for is the sharp curvature of the pulmonary and aortic valve annulus (Figure 8.1), as we currently make the assumption of a smooth continuous contour for the mitral valve annulus.

8.2 Ultrasound Image Enhancement

In this work we have presented a method to improve the temporal resolution of standard 3D echocardiography acquisitions, which included a fast image-based rigid registration method to stabilize images across a sequence and compensate for respiration and other small motions of the probe relative to the anatomy of interest. Using these methods, acquisitions acquired at about 10Hz could be reconstructed to an effective frame rate of

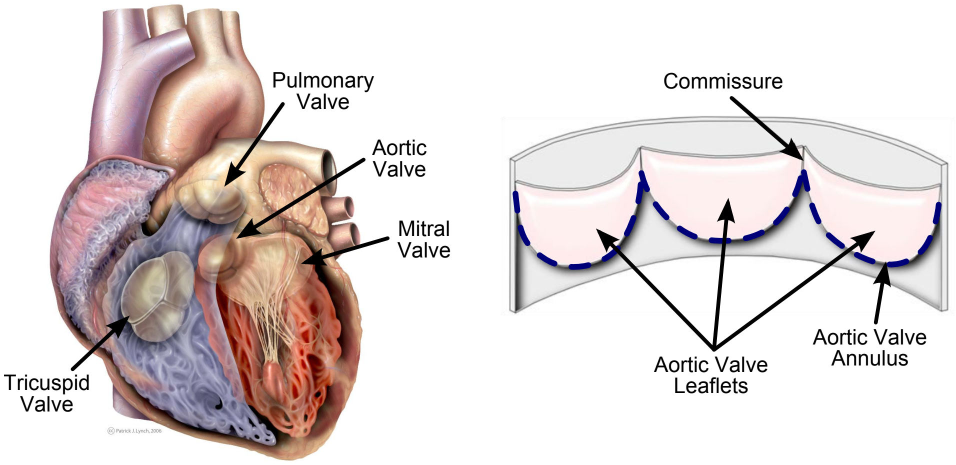


Figure 8.1: (*Left*) Depictions of the four valves in the heart (Image source: Patrick J. Lynch, medical illustrator; C. Carl Jaffe, MD, cardiologist). (*Right*) Depiction of the aortic valve cut axially between leaflets and unrolled (Image courtesy of Peter Hammer, PhD). To segment the aortic valve using the mitral valve segmentation methods, we would have to account for the sharp curvature in the aortic annulus (dashed line), as we currently make the assumption of a smooth continuous contour for the mitral annulus.

nearly 500Hz, and acquisitions could be acquired without putting patients on breath-hold or using respiration gating. Being able to both temporally and spatially register 3DUS images makes research in other areas of image enhancement of 3DUS possible. These areas are summarized below.

Whole Heart Ultrasound Imaging

Imaging the entire heart would be valuable for diagnosis and to study cardiac behavior to determine correlations between behavior and pathology. However, this is currently only possible with 3DUS using low spatial and temporal resolution imaging. Otherwise, the involved process of constructing a freehand 3DUS volume from externally tracked 2D ultrasound acquisitions is required. Being able to both temporally and spatially register small field-of-view, fine spatial resolution 3DUS images of the heart, and allowing clinicians to visualize progress using our real-time methods, a large spatial and temporal resolution image of the heart could be constructed.

Denoising

We briefly showed in Chapter 6 how registering multiple 3DUS images of the same anatomy can be used with an averaging compositing method to denoise an image, although the study we showed was not directed toward denoising 3DUS images. Therefore, images could be acquired purposefully and registered using our image-based method to better show the benefit of denoising in ultrasound. Additionally, a more sophisticated compositing method (other than simple averaging) that takes into consideration the Rayleigh distribution of ultrasound noise could be used to better denoise images [81].

Super Resolution

By registering multiple images of the same anatomy both in time and space, the same anatomy is imaged repeatedly but from slightly different positions and orientations. Therefore, it is reasonable to assume that this oversampling of a region of interest could be used to construct super-resolution images of cardiac anatomy. This could prove to be a valuable approach for the challenging problem of imaging and segmenting accurate mitral chordal structure from 3DUS.

Multiview Compositing

One of the challenges of working with ultrasound is that the appearance of anatomy is dependent on the orientation of anatomy to the acoustic propagation. The most visible structures are thicker tissue or tissue that is perpendicular to the direction of acoustic propagation. We use this to our advantage in our mitral annulus and leaflet segmentation methods. In acquiring images for these studies using an *en face* or apex approach, the mitral leaflets are visible but the chordal structure is not, allowing us to concentrate our efforts on just the annulus and leaflets. However, if we are able to accurately visualize the chords using a short-axis approach (i.e. a probe orientation that is perpendicular to the left ventricular long axis) in combination with denoising and/or super-resolution methods, it will be important to be able to register the views of the mitral leaflets and mitral chords to the same coordinate space so that a complete segmentation of the mitral valve can be generated. The spatial and temporal registration methods necessary for this could be accomplished using our designed methods.

Tissue versus Blood Segmentation

Due to noise and artifacts in ultrasound imaging, it is often difficult to determine the boundary between blood and tissue. Being able to perform denoising, and using an inhomogeneity correction method [108], segmenting tissue from blood could be made easier. This could prove useful in chordal segmentation, and might help to improve the annulus and leaflet segmentation efforts, both of which rely on a thin tissue detector (TTD) image. The TTD could be replaced with an image that is a measure of the tissue thickness as found from the blood versus tissue segmentation.

High Dynamic Range Imaging

The appearance of anatomy in ultrasound images is dependent on several different ultrasound settings, including the acoustic wave power, time and lateral gain compensation, compression, and transfer function gain. In particular, we have found that using a high power setting allows for the visualization of small structures but “blows out” the appearance of larger structures in that their internal texture is lost. Conversely, using a low power setting, larger structures are better defined but small structures are not visible. It would therefore be beneficial to construct images with a high dynamic range by registering images taken at different power settings and compositing the images using high dynamic range methods such as those used for 2D photos. A preliminary study in high dynamic range ultrasound imaging was conducted by Alice Wang as a senior thesis project at Harvard College. This study showed promise using water tank images, but this area of research warrants further investigation using clinical images.

8.3 Real-Time Feedback of Offline Segmentations

The mitral valve segmentation methods are designed for offline processing and, other than the real-time image-based rigid registration algorithm, are not designed to operate in real-time. This is generally due to the fact that the methods are designed to iterate over several possible solutions to the segmentation problem until an optimal solution is found. We found this was necessary to generate a robust and accurate segmentation. However, the methods could be made to operate in clinically feasible time (i.e. in less than 30 minutes for most standard acquisitions). It therefore would become possible in an intervention case to take a preoperative image, compute the mitral valve segmentations offline, and then feedback the segmentation data to the clinician by overlaying the segmentations on top of real-time data using the presented ultrasound registration method. This could prove especially useful for beating heart interventions such as those studied in [112].

Bibliography

[1] R.M. Ahmad, A.M. Gillinov, P.M. McCarthy, E.H. Blackstone, C. Apperson-Hansen, J.X. Qin, D. Agler, T. Shiota, and D.M. Cosgrove. Annular geometry and motion in human ischemic mitral regurgitation: novel assessment with three-dimensional echocardiography and computer reconstruction. *The Annals of Thoracic Surgery*, 78(6):2063–2068, 2004.

[2] H. Alkadhi, L. Desbiolles, P. Stolzmann, S. Leschka, H. Scheffel, A. Plass, T. Schertler, P.T. Trindade, M. Genoni, P. Cattin, et al. Mitral Annular Shape, Size, and Motion in Normals and in Patients With Cardiomyopathy: Evaluation With Computed Tomography. *Investigative Radiology*, 44(4):218–225, 2009.

[3] KS Arun, T.S. Huang, and S.D. Blostein. Least-squares fitting of two 3-D point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):698–700, 1987.

[4] J.M. Blackall, D. Rueckert, C.R. Maurer Jr, G.P. Penney, D.L.G. Hill, and D.J. Hawkes. An Image Registration Approach to Automated Calibration for Freehand 3D Ultrasound. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2000*, pages 462–471. Springer-Verlag, 2000.

[5] E.M. Boctor, A. Jain, M.A. Choti, R.H. Taylor, and G. Fichtinger. Rapid calibration method for registration and 3D tracking of ultrasound images using spatial localizer. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 5035, pages 521–532, 2003.

[6] W. Bothe, T.C. Nguyen, D.B. Ennis, A. Itoh, C.J. Carlhäll, D.T. Lai, N.B. Ingels, and D.C. Miller. Effects of acute ischemic mitral regurgitation on three-dimensional mitral leaflet edge geometry. *European Journal of Cardio-Thoracic Surgery*, 33(2):191–197, 2008.

[7] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004.

[8] S. Brekke, S.I. Rabben, A. Støylen, A. Haugen, G.U. Haugen, E.N. Steen, and H. Torp. Volume stitching in three-dimensional echocardiography: Distortion analysis and extension to real time. *Ultrasound in Medicine & Biology*, 33(5):782–796, 2007.

[9] S. Brekke, E. Tegnander, HG Torp, and SH Eik-Nes. Tissue Doppler gated (TDOG) dynamic three-dimensional ultrasound imaging of the fetal heart. *Ultrasound in Obstetrics and Gynecology*, 24(2):192–198, 2004.

[10] Philippe Burlina, Chad Sprouse, Daniel DeMenthon, Anne Jorstad, Radford Juang, Francisco Contijoch, Theodore Abraham, David Yuh, and Elliot McVeigh. Patient-Specific Modeling and Analysis of the Mitral Valve Using 3D-TEE. In *Information Processing in Computer-Assisted Interventions*, volume 6135 of *Lecture Notes in Computer Science*, pages 135–146. Springer, 2010.

[11] F. Cademartiri and P. Pavone. Advantages of retrospective ECG-gating in cardiothoracic imaging with 16-row multislice computed tomography. *Acta Biomed Ateneo Parmense*, 74:126–30, 2003.

[12] F. Cen, Y. Jiang, Z. Zhang, HT Tsui, TK Lau, and H. Xie. Robust registration of 3-D ultrasound images based on gabor filter and mean-shift method. *Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis*, LNCS 3117:304–316, 2004.

[13] P Dagum, T Timek, GR Green, GT Daughters, D Liang, NB Ingels, and DC Miller. Three-dimensional geometric comparison of partial and complete flexible mitral annuloplasty rings. *The Journal of Thoracic and Cardiovascular Surgery*, 122(4):665–673, 2001.

[14] M. Daimon, G. Saracino, A.M. Gillinov, Y. Koyama, S. Fukuda, J. Kwan, J.M. Song, V. Kongsaerepong, D.A. Agler, J.D. Thomas, et al. Local dysfunction and asymmetrical deformation of mitral annular geometry in ischemic mitral regurgitation: a novel computerized 3D echocardiographic analysis. *Echocardiography*, 25(4):414–423, 2008.

[15] D. Decarlo and D. Metaxas. Optical flow constraints on deformable models with applications to face tracking. *International Journal of Computer Vision*, 38(2):99–127, 2000.

[16] M. Drangova, N.L. Ford, S.A. Detombe, A.R. Wheatley, and D.W. Holdsworth. Fast Retrospectively Gated Quantitative Four-Dimensional (4D) Cardiac Micro Computed Tomography Imaging of Free-Breathing Mice. *Investigative Radiology*, 42(2):85–94, 2007.

[17] DR Einstein, KS Kunzelman, PG Reinhall, MA Nicosia, and RP Cochran. Non-linear fluid-coupled computational model of the mitral valve. *The Journal of Heart Valve Disease*, 14(3):376–385, 2005.

[18] Y. Eto, H. Yamada, J.H. Shin, D.A. Agler, H. Tsujino, J.X. Qin, G. Saracino, N.L. Greenberg, J.D. Thomas, and T. Shiota. Automated mitral annular tracking: A novel method for evaluating mitral annular motion using two-dimensional echocardiography. *Journal of the American Society of Echocardiography*, 18(4):306–312, 2005.

[19] A.M. Fabricius, T. Walther, V. Falk, and F.W. Mohr. Three-dimensional echocardiography for planning of mitral valve surgery: Current applicability? *The Annals of Thoracic Surgery*, 78(2):575–578, 2004.

[20] M.A. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[21] F.A. Flachskampf, S. Chandra, A. Gaddipatti, R.A. Levine, A.E. Weyman, W. Ameling, P. Hanrath, and J.D. Thomas. Analysis of shape and motion of the mitral annulus in subjects with and without cardiomyopathy by echocardiographic 3-dimensional reconstruction. *Journal of the American Society of Echocardiography*, 13(4):277–287, 2000.

[22] L. Ford and D. Fulkerson. *Flows in networks*. Princeton University Press, 1962.

[23] R. Francois, R. Fablet, and C. Barillot. Robust statistical registration of 3D ultrasound images using texture information. In *IEEE Int. Conf. on Image Processing*, volume 3, pages I–581–4. Citeseer, 2003.

[24] A.H. Gee, G.M. Treece, R.W. Prager, CJC Cash, L.H. Berman, and T. Street. Rapid registration for wide field-of-view freehand 3d ultrasound. *IEEE Transactions on Medical Imaging*, 22(11):1344–1357, 2003.

[25] JR Glasson, M Komeda, GT Daughters, MA Niczyporuk, AF Bolger, NB Ingels, and DC Miller. Three-dimensional regional dynamics of the normal mitral anulus during left ventricular ejection. *The Journal of Thoracic and Cardiovascular Surgery*, 111(3):574–585, 1996.

[26] M.E. Goldman, F. Mora, T. Guarino, V. Fuster, and B.P. Mindich. Mitral valvuloplasty is superior to valve replacement for preservation of left ventricular function: An intraoperative two-dimensional echocardiography study. *Journal of the American College of Cardiology*, 10(3):568–575, 1987.

[27] J.H. Gorman, K.B. Gupta, J.T. Streicher, R.C. Gorman, B.M. Jackson, M.B. Ratcliffe, D.K. Bogen, and L.H. Edmunds. Dynamic three-dimensional imaging of the mitral valve and left ventricle by rapid sonomicrometry array localization. *The Journal of Thoracic and Cardiovascular Surgery*, 112(3):712–726, 1996.

[28] J.S. Grashow, A.P. Yoganathan, and M.S. Sacks. Biaixal stress–stretch behavior of the mitral valve anterior leaflet at physiologic strain rates. *Annals of Biomedical Engineering*, 34(2):315–325, 2006.

[29] V. Grau, H. Becher, and J.A. Noble. Registration of multiview real-time 3-D echocardiographic sequences. *IEEE Transactions on Medical Imaging*, 26(9):1154–1165, 2007.

[30] V. Grau and J.A. Noble. Adaptive multiscale ultrasound compounding using phase information. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2005*, pages 589–596, 2005.

[31] S.R. Gunn. On the discrete representation of the Laplacian of Gaussian. *Pattern Recognition*, 32(8):1463–1472, 1999.

[32] P. Hammer, N. Vasilyev, D. Perrin, P. del Nido, and R. Howe. Fast image-based model of mitral valve closure for surgical planning. In *MICCAI 2008 Workshop Proceedings: Computational Biomechanics for Medicine III*, pages 15–26, New York, NY, 2008.

[33] P.E. Hammer, D.P. Perrin, J. Pedro, and R.D. Howe. Image-based mass-spring model of mitral valve closure for surgical planning. In *Proc. SPIE - Medical Imaging*, volume 6918, pages 69180Q1–8, 2008.

[34] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision.* Cambridge University Press New York, NY, USA, 2nd edition, 2003.

[35] M.C. Herregods, A. Tau, A. Vandeplas, B. Bijnens, and F. Van de Werf. Values for mitral valve annulus dimensions in normals and patients with mitral regurgitation. *Echocardiography*, 14(6 Pt 1):529–34, 1997.

[36] B.K.P. Horn and B.G. Schunck. Determining Optical Flow. *Artificial Intelligence*, 17:185–203, 1981.

[37] R.I. Ionasec, I. Voigt, B. Georgescu, Y. Wang, H. Houle, F. Vega-Higuera, N. Navab, and D. Comaniciu. Patient-Specific Modeling and Quantification of the Aortic and Mitral Valves From 4-D Cardiac CT and TEE. *Medical Imaging, IEEE Transactions on*, 29(9):1636–1651, 2010.

[38] J.H. Jimenez, D.D. Soerensen, Z. He, S. He, and A.P. Yoganathan. Effects of a saddle shaped annulus on mitral valve function and chordal force distribution: an in vitro study. *Annals of Biomedical Engineering*, 31(10):1171–1181, 2003.

[39] K. Johnson. Introduction to rodent cardiac imaging. *ILAR Journal*, 49(1):27–34, 2008.

[40] S. Kaji, M. Nasu, A. Yamamuro, K. Tanabe, K. Nagai, T. Tani, K. Tamita, K. Shiratori, M. Kinoshita, M. Senda, et al. Annular Geometry in Patients With Chronic Ischemic Mitral Regurgitation: Three-Dimensional Magnetic Resonance Imaging Study. *Circulation*, 112(9 Suppl.):I409–I414, 2005.

[41] S.R. Kaplan, G. Bashein, F.H. Sheehan, M.E. Legget, B. Munt, X.N. Li, M. Sivarajan, E.L. Bolson, M.Z.M. Arch, and R.W. Martin. Three-dimensional echocardiographic assessment of annular shape changes in the normal and regurgitant mitral valve. *American Heart Journal*, 139(3):378–387, 2000.

[42] K. Karadayi, T. Hayashi, and Y. Kim. Automatic image-based gating for 4D ultrasound. In *Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE*, pages 2388–2391. IEEE, 2008.

[43] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.

[44] D.T. Kettler, R.D. Plowes, P.M. Novotny, N.V. Vasilyev, P.J. del Nido, and RD Howe. An active motion compensation instrument for beating heart mitral valve surgery. In

Proc. IEEE International Conference on Intelligent Robots and Systems, pages 1290–1295, 2007.

[45] V. Kolmogorov. Software, 2004. http://www.cs.ucl.ac.uk/staff/V.Kolmogorov/-software.html. [Accessed: October 1, 2008].

[46] K. S. Kunzelman, R. P. Cochran, C. Chuong, W. S. Ring, E. D. Verrier, and R. D. Eberhart. Finite element analysis of the mitral valve. *The Journal of Heart Valve Disease*, 2(3):326–40, 1993.

[47] KS Kunzelman, DR Einstein, and RP Cochran. Fluid–structure interaction models of the mitral valve: function in normal and pathological states. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 362(1484):1393, 2007.

[48] O. Kutter, W. Wein, and N. Navab. Multi-modal Registration Based Ultrasound Mosaicing. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2009*, pages 763–770, 2009.

[49] Z. Lei, Y. Xin, Y. Liping, and S. Kun. Three Dimensional Reconstruction and Dynamic Analysis of Mitral Annular Based on Connected Equi-length Curve Angle Chain. In *Lecture Notes in Computer Science - International Conference on Medical Biometrics*, volume 4901, pages 298–306. Springer, 2007.

[50] G.W. Lenz, E.M. Haacke, and R.D. White. Retrospective cardiac gating: a review of technical aspects and future directions. *Magnetic Resonance Imaging*, 7(5):445–455, 1989.

[51] RA Levine, MD Handschumacher, AJ Sanfilippo, AA Hagege, P. Harrigan, JE Marshall, and AE Weyman. Three-dimensional echocardiographic reconstruction of the mitral valve, with implications for the diagnosis of mitral valve prolapse. *Circulation*, 80(3):589–598, 1989.

[52] K.H. Lim, J.H. Yeo, and CM Duran. Three-dimensional asymmetrical modeling of the mitral valve: a finite element study with dynamic boundaries. *The Journal of Heart Valve Disease*, 14(3):386–392, 2005.

[53] S. Loncaric and T. Macan. Point-constrained optical flow for LV motion detection. In *Proceedings of SPIE*, volume 3978, pages 521–529, 2000.

[54] Y. Looney and P. Quinton. Mitral Valve Surgery. *Continuing Education in Anaesthesia, Critical Care & Pain*, 5(6):199, 2005.

[55] A.M. López, F. Lumbreras, J. Serrat, and J.J. Villanueva. Evaluation of methods for ridge and valley detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4):327–335, 1999.

[56] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[57] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. 7th International Joint Conference on Artificial Intelligence*, pages 674–679, Vancouver, B.C., Canada, 1981.

[58] S. Martin, V. Daanen, O. Chavanon, and J. Troccaz. Fast Segmentation of the Mitral Valve Leaflet in Echocardiography. *Lecture Notes in Computer Science - Computer Vision Approaches to Medical Image Analysis*, 4241:225–235, 2006.

[59] L. Mercier, T. Langu, F. Lindseth, and L.D. Collins. A review of calibration techniques for freehand 3-D ultrasound systems. *Ultrasound in Medicine & Biology*, 31(2):143–165, 2005.

[60] I. Mikic, S. Krucinski, and J.D. Thomas. Segmentation and tracking in echocardiographic sequences: Active contours guided by optical flow estimates. *Medical Imaging, IEEE Transactions on*, 17(2):274–284, 1998.

[61] K. Mikolajczyk. Detection of local features invariant to affine transformations. *PhD Thesis, Institut National Polytechnique de Grenoble, France*, 2002.

[62] ME Miquel, RS Razavi, EJ Baker, DLG Hill, and SF Keevil. 3D Reconstruction of Intra-Cardiac Anatomy: A Comparison of Magnetic Resonance Imaging (MRI), Intra-Cardiac Echocardiography (ICE) and Trans-Esophagal Echocardiography (TEE). In *Proc Intl Soc Mag Reson Med*, volume 9, page 1824, 2001.

[63] M. Moradi, P. Abolmaesoumi, and P. Mousavi. Deformable registration using scale space keypoints. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 6144, pages 791–798, 2006.

[64] R.R. Moss, K.H. Humphries, M. Gao, C.R. Thompson, J.G. Abel, G. Fradet, and B.I. Munt. Outcome of mitral valve repair or replacement: a comparison by propensity score analysis. *Circulation*, 108(90101):II–90–II–97, 2003.

[65] H. Neemuchwala, A. Hero, and P. Carson. Image matching using alpha-entropy measures and entropic graphs. *Signal Processing*, 85(2):277–296, 2005.

[66] S.T. Nevo, M. van Stralen, A.M. Vossepoel, J.H.C. Reiber, N. de Jong, A.F.W. van der Steen, and J.G. Bosch. Automated tracking of the mitral valve annulus motion in apical echocardiographic images using multidimensional dynamic programming. *Ultrasound in Medicine & Biology*, 33(9):1389–1399, 2007.

[67] D. Ni, Y. Qu, X. Yang, Y. Chui, T.T. Wong, S. Ho, and P. Heng. Volumetric ultrasound panorama based on 3D SIFT. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2008*, pages 52–60, 2008.

[68] V.T. Nkomo, J.M. Gardin, T.N. Skelton, J.S. Gottdiener, C.G. Scott, and M. Enriquez-Sarano. Burden of valvular heart diseases: a population-based study. *The Lancet*, 368(9540):1005–1011, 2006.

[69] D. Pace, D. Gobbi, C. Wedlake, J. Gumprecht, J. Boisvert, J. Tokuda, N. Hata, and T. Peters. An Open-source Real-time Ultrasound Reconstruction System for Four-dimensional Imaging of Moving Organs. *Med Image Comput Comput Assist Interv. MICCAI 2009*, 09 2009.

[70] GP Penney, JM Blackall, MS Hamady, T. Sabharwal, A. Adam, and DJ Hawkes. Registration of freehand 3D ultrasound and magnetic resonance liver images. *Medical Image Analysis*, 8(1):81–91, 2004.

[71] T.C. Poon and R.N. Rohling. Three-dimensional extended field-of-view ultrasound. *Ultrasound in Medicine & Biology*, 32(3):357–369, 2006.

[72] Z.B. Popovic, M. Martin, K. Fukamachi, M. Inoue, J. Kwan, K. Doi, J.X. Qin, T. Shiota, M.J. Garcia, P.M. McCarthy, et al. Mitral annulus size links ventricular dilatation to functional mitral regurgitation. *Journal of the American Society of Echocardiography*, 18(9):959–963, 2005.

[73] B.C. Porter. *Three-dimensional medical ultrasound acquisition and data registration and fusion.* PhD thesis, University of Rochester, 2004.

[74] J.X. Qin, T. Shiota, H. Tsujino, G. Saracino, R.D. White, N.L. Greenberg, J. Kwan, Z.B. Popovic, D.A. Agler, W.J. Stewart, et al. Mitral annular motion as a surrogate for left ventricular ejection fraction: real-time three-dimensional echocardiography and magnetic resonance imaging studies. *European Journal of Echocardiography*, 5(6):407–415, 2004.

[75] K. Rajpoot, J.A. Noble, V. Grau, C. Szmigielski, and H. Becher. Multiview RT3D Echocardiography Image Fusion. In *Proceedings of the 5th International Conference on Functional Imaging and Modeling of the Heart*, pages 134–143. Springer-Verlag, 2009.

[76] S. Ratanasopa, E.L. Bolson, F.H. Sheehan, J.A. McDonald, and G. Bashein. Performance of a Fourier-based program for three-dimensional reconstruction of the mitral annulus on application to sparse, noisy data. *The International Journal of Cardiac Imaging*, 15(4):301–307, 1999.

[77] R.N. Rohling, AH Gee, and L. Berman. Automatic registration of 3-D ultrasound images. *Ultrasound in Medicine & Biology*, 24(6):841–854, 1998.

[78] S. Roy. Stereo without epipolar lines: A maximum-flow formulation. *International Journal of Computer Vision*, 34(2):147–161, 1999.

[79] I.S. Salgo, J.H. Gorman, R.C. Gorman, B.M. Jackson, F.W. Bowen, T. Plappert, M.G. St John Sutton, and L.H. Edmunds. Effect of annular shape on leaflet curvature in reducing mitral leaflet stress. *Circulation*, 106(6):711–717, 2002.

[80] G. Saracino, M. Daimon, NL Greenberg, T. Shiota, and JD Thomas. A Novel System for the Assessment of Mitral Annular Geometry and Analysis of 3D Motion of Mitral Annulus from 3D Echocardiography. *Computers in Cardiology*, 31:69–72, 2004.

[81] A. Sarti, C. Corsi, E. Mazzini, and C. Lamberti. Maximum likelihood segmentation of ultrasound images with Rayleigh distribution. *Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on*, 52(6):947–960, 2005.

[82] R.J. Schneider, D.P. Perrin, N.V. Vasilyev, G.R. Marx, P.J. del Nido, and R.D. Howe. Mitral annulus segmentation from three-dimensional ultrasound. In *Proc. IEEE International Symposium on Biomedical Imaging*, pages 779–82, Boston, MA, 2009. IEEE.

[83] R.J. Schneider, D.P. Perrin, N.V. Vasilyev, G.R. Marx, P.J. del Nido, and R.D. Howe. Mitral Annulus Segmentation from 3D Ultrasound Using Graph Cuts. *IEEE Transactions on Medical Imaging*, 29(9):1676–87, 2010.

[84] J.A. Sethian. *Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science.* Cambridge University Press, 1999.

[85] Y. Shang, X. Yang, L. Zhu, R. Deklerck, and E. Nyssen. Region competition based active contour for medical object extraction. *Computerized Medical Imaging and Graphics*, 32(2):109–117, 2008.

[86] R. Shekhar, V. Zagrodsky, C.R. Castro-Pareja, V. Walimbe, and J.M. Jagadeesh. High-Speed Registration of Three-and Four-dimensional Medical Images by Using Voxel Similarity. *Radiographics*, 23(6):1673, 2003.

[87] C. Sheng, Y. Xin, Y. Liping, and S. Kun. Segmentation in echocardiographic sequences using shape-based snake model combined with generalized Hough transformation. *The International Journal of Cardiovascular Imaging (formerly Cardiac Imaging)*, 22(1):33–45, 2006.

[88] O.V. Solberg, F. Lindseth, H. Torp, R.E. Blake, and T.A. Nagelhus Hernes. Freehand 3D ultrasound reconstruction algorithms–a review. *Ultrasound in Medicine & Biology*, 33(7):991–1009, 2007.

[89] P. Soler, O. Gerard, P. Allain, E. Saloux, E. Angelini, and I. Bloch. Comparison of fusion techniques for 3D+T echocardiography acquisitions from different acoustic windows. *Computers in Cardiology, 2005*, 32:141–144, 2005.

[90] Z. Sun. Multislice CT angiography in cardiac imaging: prospective ECG-gating or retrospective ECG-gating? *Biomedical Imaging and Intervention Journal*, 6(1):e4, 2010.

[91] Y. Takemoto, T. Hozumi, K. Sugioka, H. Watanabe, Y. Matsumura, M. Yoshiyama, K. Takeuchi, and J. Yoshikawa. Automated Three-Dimensional Analysis of Mitral Annular Dynamics in Patients with Myocardial Infarction Using Automated Mitral Annular Tracking Method. *Echocardiography*, 23(8):658–665, 2006.

[92] D. Terzopoulos, A. Witkin, and M. Kass. Constraints on deformable models: Recovering 3D shape and nonrigid motion. *Artificial Intelligence*, 36(1):91–123, 1988.

[93] TA Timek, P Dagum, DT Lai, D Liang, GT Daughters, F Tibayan, NB Ingels, and DC Miller. Tachycardia-induced cardiomyopathy in the ovine heart: Mitral annular dynamic three-dimensional geometry. *The Journal of Thoracic and Cardiovascular Surgery*, 125(2):315–324, 2003.

[94] M.D. Tischler, K.A. Cooper, M. Rowen, and M.M. LeWinter. Mitral valve replacement versus mitral valve repair. A Doppler and quantitative stress echocardiographic study. *Circulation*, 89(1):132–137, 1994.

[95] Z.G. Turi. Mitral valve disease. *Circulation*, 109(6):e38–e41, 2004.

[96] MT Upton, DG Gibson, and DJ Brown. Instantaneous mitral valve leaflet velocity and its relation to left ventricular wall movement in normal subjects. *British Heart Journal*, 38(1):51–58, 1976.

[97] M. Van Stralen, KYE Leung, MM Voormolen, N. De Jong, AFW Van der Steen, JHC Reiber, and JG Bosch. Time Continuous Detection of the Left Ventricular Long Axis and the Mitral Valve Plane in 3-D Echocardiography. *Ultrasound in Medicine & Biology*, 34(2):196–207, 2008.

[98] F. Veronesi, C. Corsi, EG Caiani, L. Sugeng, L. Weinert, V. Mor-Avi, RM Lang, and C. Lamberti. Semi-automatic tracking for mitral annulus dynamic analysis using real-time 3D echocardiography. *Computers in Cardiology*, 33:113–116, 2006.

[99] I. Voigt, R.I. Ionasec, B. Georgescu, H. Houle, M. Huber, J. Hornegger, and D. Comaniciu. Model-driven physiological assessment of the mitral valve from 4D TEE. In *Proc. SPIE - Medical Imaging*, volume 7261, page 72610R (11pp.). SPIE, 2009.

[100] E. Votta, E. Caiani, F. Veronesi, M. Soncini, F.M. Montevecchi, and A. Redaelli. Mitral valve finite-element modelling from ultrasound data: a pilot study for a new approach to understand mitral function and clinical scenarios. *Philosophical Transactions of the Royal Society London, Series A*, 366(1879):3411–3434, 2008.

[101] C. Wachinger, W. Wein, and N. Navab. Registration Strategies and Similarity Measures for Three-dimensional Ultrasound Mosaicing. *Academic Radiology*, 15(11):1404–1415, 2008.

[102] C.R. Wagner and D.P. Perrin. Efficient curvature estimations for real-time (25Hz) segmentation of volumetric ultrasound data. In *Proc. SPIE - Medical Imaging*, volume 6914, pages 1–10. SPIE, 2008.

[103] Z. Wang, G. Slabaugh, G. Unal, and T. Fang. Registration of Ultrasound Images using an Information-Theoretic Feature Detector. In *Proc. IEEE International Symposium on Biomedical Imaging*, pages 736–739, Washington, D.C., 2007. IEEE.

[104] N. Watanabe, Y. Ogasawara, Y. Yamaura, K. Yamamoto, N. Wada, N. Okahashi, T. Kawamoto, E. Toyota, and K. Yoshida. Dynamics of mitral complex geometry and functional mitral regurgitation during heart failure treatment. *Journal of Echocardiography*, 4(2):51–58, 2006.

[105] W. Wein, E. Camus, M. John, M. Diallo, C. Duong, A. Al-Ahmad, R. Fahrig, A. Khamene, and C. Xu. Towards Guidance of Electrophysiological Procedures with Real-Time 3D Intracardiac Echocardiography Fusion to C-arm CT. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2009*, pages 9–16, 2009.

[106] D.J. Williams and M. Shah. A fast algorithm for active contours and curvature estimation. *CVGIP: Image Understanding*, 55(1):14–26, 1992.

[107] I. Wolf, M. Hastenteufel, R. De Simone, C. Vahl, S. Hagl, and HP Meinzer. Three-dimensional annulus segmentation and hybrid visualisation in echocardiography. *Computers in Cardiology*, pages 105–108, 2001.

[108] G. Xiao, J.M. Brady, J.A. Noble, M. Burcher, and R. English. Nonrigid registration of 3-D free-hand ultrasound images of the breast. *IEEE Transactions on Medical Imaging*, 21(4):405, 2002.

[109] C. Yao, J.M. Simpson, C.H.P. Jansen, A.P. King, and G.P. Penney. Spatial compounding of large sets of 3D echocardiography images. In *Proc. SPIE - Medical Imaging*, volume 7265, pages 726515–1–8, 2009.

[110] H. Yu, M.S. Pattichis, C. Agurto, and M.B. Goens. A 3D Freehand Ultrasound System for Multi-view Reconstructions from Sparse 2D Scanning Planes. *BioMedical Engineering OnLine*, 10(1):7, 2011.

[111] H. Yu, M.S. Pattichis, and M.B. Goens. Multi-view 3D reconstruction with volumetric registration in a freehand ultrasound imaging system. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 6147, pages 45–56. Citeseer, 2006.

[112] S.G. Yuen, D.P. Perrin, N.V. Vasilyev, P.J. del Nido, and R.D. Howe. Force tracking with feed-forward motion estimation for beating heart surgery. *IEEE Transactions on Robotics*, 26(5):888–896, 2010.

[113] Z. Zhang and M. Braun. Fully 3D active surface models with self-inflation and self-deflation forces. In *Computer Vision and Pattern Recognition*, pages 85–90. IEEE, 2002.

[114] X. Zhuang, C. Yao, Y. Ma, D. Hawkes, G. Penney, and S. Ourselin. Registration-based propagation for whole heart segmentation from compounded 3d echocardiography. In *Proc. IEEE International Symposium on Biomedical Imaging*, pages 1093–96, Rotterdam, The Netherlands, 2010. IEEE.