

Techniques for Quantitation of Left Ventricular Volume in Ultrasound Using 4DViz

by

Yuan Guo

Department of Biomedical Engineering  
Duke University

Date: \_\_\_\_\_

Approved:

\_\_\_\_\_  
Olaf T. von Ramm, Supervisor

\_\_\_\_\_  
William M. Reichert

\_\_\_\_\_  
Adam P. Wax

Thesis submitted in partial fulfillment of  
the requirements for the degree of Master of Science in the Department of  
Biomedical Engineering in the Graduate School  
of Duke University

2012

ABSTRACT

Techniques for Quantitation of Left Ventricular Volume in Ultrasound Using 4DViz

by

Yuan Guo

Department of Biomedical Engineering  
Duke University

Date: \_\_\_\_\_

Approved:

\_\_\_\_\_  
Olaf T. von Ramm, Supervisor

\_\_\_\_\_  
William M. Reichert

\_\_\_\_\_  
Adam P. Wax

An abstract of a thesis submitted in partial  
fulfillment of the requirements for the degree  
of Master of Science in the Department of  
Biomedical Engineering in the Graduate School  
of Duke University

2012

Copyright by  
Yuan Guo  
2012

## Abstract

In the United States, heart failure is a leading cause of hospitalization. The medical industry places great emphasis on diagnosing heart disease through cardiac metrics like ejection fraction. Left ventricular ejection fraction is a commonly used diagnostic indicator for heart efficiency and is measured with echocardiography through different volume calculation techniques. However, ejection fraction results can drastically vary from one examiner to another. Generally cardiologists still give ejection fraction measurements a plus or minus 10 percent error range.

A program developed at Duke called 4DViz is robust enough for users to process 3D ultrasound data. 4DViz allows examiners to determine heart chamber volumes by constructing a surface model over an imaged heart chamber with many mouse click inputs. Through 4DViz programming, a viable approach for calculating ejection fraction is attempted in this thesis. Using feature tracking, surface drawing, and voxel filling, the new approach aims to reduce examiner input and improve ejection fraction consistency. Water filled balloons were used to calibrate the algorithm's parameters. In testing, several volunteers were asked to use the 4DViz. Their results are compared to volume measurements where user input was standard. The results show promise and may remove some of the inconsistency behind ejection fraction measurements.

# Contents

Abstract .....	iv
List of Tables .....	vii
List of Figures .....	viii
Acknowledgements .....	xi
1. Introduction and Background.....	1
1.1 Heart Disease .....	1
1.2 Cardiac Ultrasound.....	3
1.3 Left Ventricular Volume and Ejection Fraction.....	6
1.4 Volume Measurement Techniques and Variability .....	7
2. 4DViz .....	9
2.1 Display Modes and Data .....	10
2.2 Surface Reconstruction Model.....	12
2.3 Initial Applications in Volume Charts.....	14
2.4 Hypothesis and Methods .....	16
3. Applying Water Balloon Data and Heart Data.....	17
3.1 Balloon Imaging.....	18
3.2 Volume Data Sets of Hearts .....	19
3.3 Challenges of the Ventricle Wall in Ultrasound .....	20
4. The Filler Algorithm Techniques.....	21
4.1 Applying a Threshold to the Data.....	22

4.2 Feature Searching with the Surface Model .....	24
4.3 Automated Feature Searching .....	26
4.4 Sum of Absolute Difference Tracking .....	31
4.5 Calibrating SAD Threshold and Search Range .....	36
4.6 Approximating the Ventricle Surface with Lines .....	39
4.7 Calibrating Line Drawing Range .....	41
4.8 The Filler Function .....	46
4.9 Filler Algorithm Summary .....	51
5. Testing and Results.....	52
5.1 Volume Results for Each Data Set.....	53
5.2 Comparison of Algorithm Techniques.....	55
6. Conclusion .....	62
Bibliography .....	63

## List of Tables

Table 1: Two 3x3x3 SAD Histograms for 110mL balloon Showing the Effect of Thresholding.....	37
Table 2: Results of AddLine Tests on 110.0mL Balloon.....	42
Table 3: Results of AddLine search range tests on 55.6mL Balloon.....	43
Table 4: Results of AddLine search range tests on Normal Heart Image .....	44
Table 5: Volume and Ejection Fraction Results from Surface Model Technique .....	53

## List of Figures

Figure 1: Transmitted ultrasonic pulses (black) reflect back to probe (red). .....	3
Figure 2: Ejection Fraction calculations done by two examiners. Each yellow dot represents one 2D image examined by both. The diffuse arrangement of the dots suggests a substantial variability in measurements.....	8
Figure 3: (A) Render View of ultrasound volume data. A single slice of the left ventricle is shown. (B) The 3D surface reconstruction model applied over the ventricle on the systolic frame.....	12
Figure 4: The surface reconstruction model is cylindrical in design because the left ventricle can be modeled as a cylinder and volume can be calculated from cylinder equations. There is no way to model the planar base.....	13
Figure 5: In these ultrasound images, ventricle walls appear to be missing. This is fairly common due to the angular dependence of sound reflection. Tracing these missing ventricle walls can often be guesswork. ....	14
Figure 6: Volume charts measured by eight different students and their average overall. On many frames, a 20 milliliter range around the average shows the user dependence in producing consistent volumes on ultrasound images.....	15
Figure 7: Ultrasound images of balloons. The leftmost has 110.0mL of water, the center has 83.8mL, and the rightmost has 55.6mL. ....	18
Figure 8: The interior wall of the left ventricle lined with trabeculae. The circle highlights possible noise from trabeculae. ....	20
Figure 9: (A) and (C) are unaltered ultrasound volumes in Render and Volume view modes, while (B) and (D) are the same but have had a regional threshold applied. In (B), the ventricle wall at the yellow arrow is gone because the threshold was skewed. Bright voxels pointed to by yellow arrows in (C) and (D) skewed the histogram.....	23
Figure 10: Feature searching with and without base features. The translated surface model is show in 2D on the left and 3D on the right (tilted with base pointing upward). The introduction of base features is shown on the bottom half. ....	25

Figure 11: A brightness chart of all voxels along the first search radius. The search range is just 1cm to 3cm in the Normal Heart data set. Ventricle chamber, trabeculation, and ventricle wall can be identified visually. .... 27

Figure 12: A derivative chart of Figure 11. The noise is easily spotted with its large rise (A) and steep fall shortly after (B). Recording the distance between a significant rise to a significant fall can identify where trabeculation and ventricle wall are located, in this case between (C) and (D). .... 28

Figure 13: The left images show the results of the automatic feature search and the right images show the original ultrasound image. A ventricle is identifiable in the automatic results, but many features are at incorrect locations as well. .... 30

Figure 14: Comparing a 3x3 square to another 3x3 square in the next frame. The sum of the absolute difference of the squares is 493. .... 31

Figure 15: An example of why low correlation thresholds are beneficial. (A) A sum of absolute difference threshold of 400 will track the feature down one pixel with a difference total that barely passes the threshold. (B) No difference total from (A) meets the new threshold of 100 so the search range is expanded in red arrows. The top left corner of the image is the optimal location to track the original feature. .... 33

Figure 16: The dummy volume, Selected Features, records local maxima from the tracking algorithm. (A) is the initial end-systolic frame. (B) is 2 frames later. (C) is 6 frames later. The organization of the features spreads out over time. .... 35

Figure 17: (A) shows the local maxima voxels on a single cut plane in Select Features. (B) shows the algorithm's use of *AddLine*. Notice that there are some gaps in the surface, but not too many and none too large. The base has no lines because the features are too far apart. In (C) the search range is doubled and the apex is now too thick. .... 40

Figure 18: (C) Filler has been applied to the surface in (B). .... 51

Figure 19: Results of algorithm using only surface model. Volumes from 5 participants are widely varied around the hand drawn average and the slopes of the curves also do not match the hand drawn average. .... 55

Figure 20: Filled ventricle chamber matches on initial Frame 13, but by Frame 1, the filled space has not kept up with the base of the ventricle as seen by the blue line. .... 56

Figure 21: Adding base features to the surface model improves the shape of the volume curve in matching the student's hand drawn average. ....57

Figure 22: Volume curves generated by the automatic algorithm for feature searching. The participant volumes are fairly consistent, but are overestimated and do not have the shape of the student tracings..... 58

Figure 23: The results of the density filter on the automated feature search. The density filter has removed much of the features outside the ventricle walls..... 59

Figure 24: The density filter adjusts the shape of the volume curves to more closely match the average of the student tracings. The consistency of the participants' volumes is improved as well. .... 60

Figure 25: Volume curves from Figure 22 have been offset by -50mL. The resulting volume curves appear to accurately measure the average tracings. .... 60

## Acknowledgements

I want to thank Dr. Olaf von Ramm for allowing me to contribute in his cardiac ultrasound lab and mentoring me these last two years. I am grateful to have learned so much from him about engineering but even more so about the world around us. I have had an excellent time working for and learning under him and his team of engineers.

I want to thank John “Lucci” Castellucci for his kind and patient mentoring. I can think of no one I’d rather work under and I’ll always appreciate his fantastic support and everlasting wit.

I want to thank Dr. Joe Kisslo for teaching me everything I know about the anatomical heart, putting a heart in my hands, and explaining why it’s important to have held a heart in my hands. He is a favorite amongst my Duke teachers and I appreciate his genuine interest in his students.

I want to thank Cooper Moore for, without him, I’d still be debugging. Quite the storyteller, he is amongst the most knowledgeable people I’ve met at Duke. 4DViz is his baby and I’m happy to have helped him out in any way.

Finally I want to thank my parents for all their love. They motivate my every action and all I hope is to make them proud.

# 1. Introduction and Background

## 1.1 Heart Disease

Heart disease is the most common cause of death for men and women in developed countries, claiming approximately a million lives annually. Hospital admission for heart failure has some of the highest mortality rates, worsening as age increases. If rates continue, heart disease will be the leading cause of death around the world by 2020 (American Heart Association, 2011).

Diagnosis and treatment of heart disease is a vast section of the biomedical industry. Prosthetic heart valves, coronary stents, pacemakers, and left ventricular assist devices are well explored treatment options for heart failure, but only if heart failure can be diagnosed before the treatment.

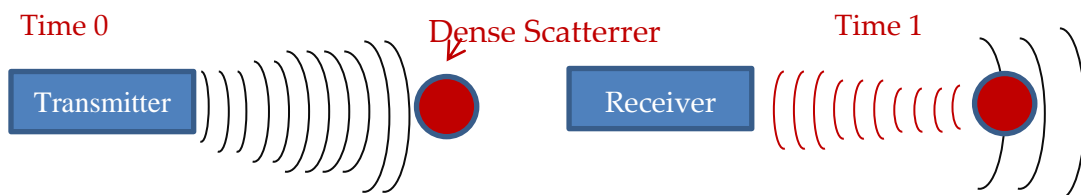
Because heart disease can come in several forms, cardiac imaging is a key diagnostic tool. Imaging the heart over even a single heartbeat can detect coronary blockage, diseased myocardium, or valve disorder. Real-time ultrasound is particularly useful for cardiologists to see motion and blood flow in heart chambers. The biomedical industry spends much effort in developing cardiac imaging modalities. Despite nearly half of heart attacks occurring without prior warning or signal, monitoring and treatment of heart disease is improving with imaging (The Heart Foundation, 2012).

Angiography, CT, MRI, nuclear imaging, and ultrasound are all capable of imaging the heart and each modality provides its own advantages and disadvantages. Although many biomedical companies offer their modalities as the “gold standard” for cardiac imaging, echocardiography is the most commonly used modality, largely based on its real-time, non-invasive, safe, and mobile application.

## 1.2 Cardiac Ultrasound

Cardiac ultrasound, also known as echocardiography, is the most widely used diagnostic tool in cardiology. The feasibility of ultrasound as a cardiac imaging application was proven in 1954 by Edler and Hertz (Fraser, 2001) in a recording of mitral valve motion from ultrasound echoes. Ultrasound is useful now in identifying heart components, motion, and measuring internal distances.

Ultrasound uses high frequency sound wave propagation and reflection in a medium to create images. Ultrasound machines use transducer probes, which are devices lined with arrays of piezoelectric elements. A piezoelectric material can bend when a voltage is applied, so applying an alternating current voltage greater than 20 kilohertz causes the piezoelectric array to vibrate and send transverse pressure waves at roughly the same frequency. The piezoelectric transducer translates voltage into pressure and vice versa, so it can both transmit and receive ultrasonic pulses. As ultrasonic pulses travel through a medium like the human body, fractions of the pulse are reflected back if there is a sudden change in acoustic impedance, a material property proportional to density. This is diagrammed in the figure below:



**Figure 1: Transmitted ultrasonic pulses (black) reflect back to probe (red).**

Imaging is possible by timing transmission and reflection. Because the speed of sound in soft human tissue is 1,540 meters per second on average, the distance of an acoustic impedance change can be calculated by timing the sound pulse's return to the transducer after transmission. A-mode ultrasound creates charts of these pulse pressure amplitudes over distance. Sequentially moving the transducer probe to create many A-modes generates the B-mode ultrasound images, where brightness is proportional to received pulse amplitude. B-mode images of the heart were the first practical application of ultrasound for diagnostic purposes.

Since then, new hardware and software have expanded ultrasound imaging. Phased piezoelectric arrays, which precisely schedule pulse transmissions for each array element, created electrically steerable wave fronts. Phased arrays replaced mechanically motioned transducers. A phased transducer has an 80 degree field of view when imaging. When phased arrays became two dimensional, 3D ultrasounds were possible, and by 1973 they have gained many clinical applications (Nelson, 1993) especially in cardiology. By timing ultrasonic slices with an EKG, ultrasound machines could acquire heart volume data over a cardiac cycle.

In the last forty years, cardiac ultrasound has added several features for diagnosing for heart disease. In 1979, Doppler imaging was integrated into ultrasound images to visually represent the direction of blood flow, useful in determining heart valve performance. By 1990, three-dimensional ultrasound imaged and displayed 3D

heart chambers and valves. Using 3D ultrasound data, many image processing techniques have developed as measuring tools: surface reconstruction (Ohazama, 1998), volume calculations, and border detection (Gorcsan, 1993). Each of these processing techniques has had to combat ultrasound's limitations in resolution and artifacts.

Ultrasound presents several challenges. For example, resolution is considerably lower than other modalities. Ultrasound's spatial resolution is closely related to the frequency and length of the ultrasonic pulses. High resolution requires high frequency, but ultrasonic waves are attenuated in tissue at a rate exponential to frequency. The standard heart is imaged at least twelve centimeters deep, so attenuation can be severe. Machines use the highest frequency allowable until attenuation limits signal-to-noise ratio (Christensen, 1988). Artifacts can be observed in the image when sound reverberates from implants or epicardium. Also, images are drastically user dependent and it can be difficult for users to angle the ultrasound probe between the lungs (low density) and ribs (high density).

Still, ultrasound has lasting advantages. Ultrasound's high frame rate can capture enough images for the cardiac cycle to appear real time. Ultrasound is also relatively safe and non-invasive. Ultrasound machines are mobile in hospitals and there are some portable prototypes. Ultrasound machines are also relatively inexpensive compared to CT and MRI.

### 1.3 Left Ventricular Volume and Ejection Fraction

Measuring volumes within the heart chambers can determine several diagnostic indicators. Volumes have previously been quantitated by removing hearts and filling chambers with fluid. A normal left ventricle contains 120 milliliters of blood when relaxed and 50 milliliters when contracted. From volumes, ejection fraction is calculable. Ejection fraction is the ratio of blood volume pumped over blood volume capacity. Mathematically, ejection fraction is calculated with two variables: end-diastolic volume and end-systolic volume. End-diastolic volume is the maximum volume of the ventricle, at its most relaxed state following systolic compression. End-systolic volume is the minimum volume of the ventricle at peak contraction. Subtracting end-systolic volume from end-diastolic volume gives stroke volume, the blood volume pumped by the ventricle per heartbeat. Stroke volume over end-diastolic volume is ejection fraction.

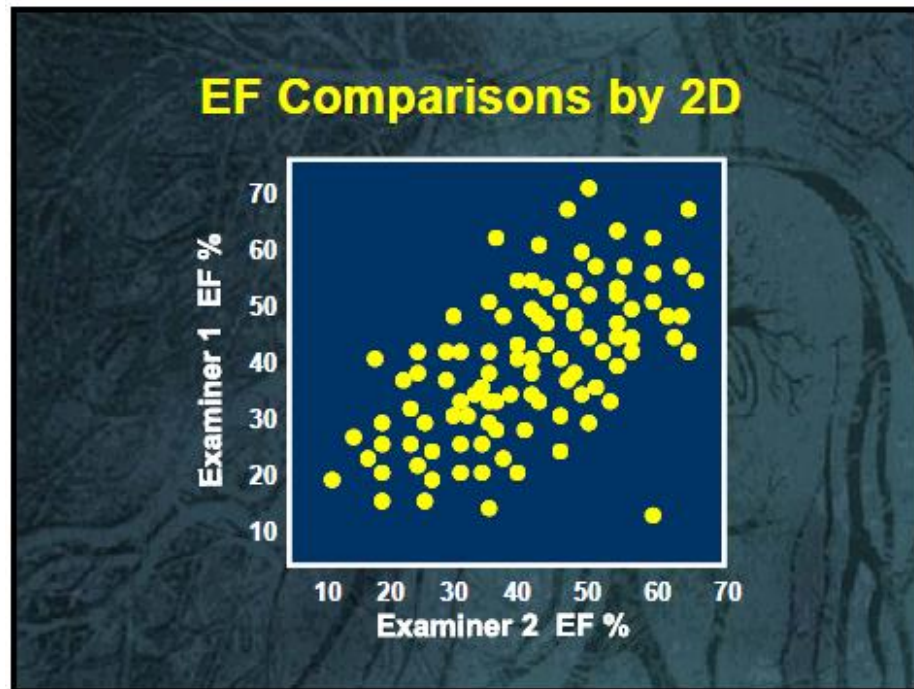
As a heart disease indicator, ejection fraction is about ranges. Cardiologists consider the normal range to be 55 to 70 percent. From 40 to 55 percent, the heart is monitored for decreasing ejection fraction. An ejection fraction below 40 percent can confirm heart disease. Falling to below 35 percent is life threatening and irregular heartbeats are common. Although it is not the sole indicator considered, the measurement of ejection fraction influences diagnosis. These ranges are moderately dependent on age, health, gender, and even the imaging modality used to measure.

## 1.4 Volume Measurement Techniques and Variability

Many image processing techniques for volume measurement simplify ventricles as geometric models. For example, if the ventricle is treated as an ellipsoid, then a volume can be acquired with a wall-to-wall diameter and valve to apex length; those distances are examined on 2D ultrasound (Craig, 1993). Simpson's Rule and modified versions have been popular geometric models for ultrasound and their resulting ejection fractions have been comparable to those acquired by x-ray cineangiography (Folland, 1979). Geometric models have had some degree of accuracy, but are reliant on the quality of ultrasound images.

Calculated ejection fractions from ultrasound images have significant variance due to examiner input. One reason for this is image quality, which often varies from one trained sonographer to another. Even with a superb image where the heart chamber is complete and distinct, the measurements on the image can drastically vary between two examiners. Figure 2 displays the difference between two examiners.

Recognizing this variance, cardiologists generally treat calculated ejection fractions with a plus or minus 10 percent error. All things considered about ultrasound imaging, a 10 percent error range is a remarkable achievement. However, the range can be problematic. If ejection fraction is calculated to 50 percent, 60 percent would be a normal ejection fraction, but 40 percent would warrant patient concern. Cardiologists would need to examine other cardiac indicators as a follow-up.



**Figure 2: Ejection Fraction calculations done by two examiners. Each yellow dot represents one 2D image examined by both. The diffuse arrangement of the dots suggests a substantial variability in measurements.**

Ejection fraction is an important cardiac indicator. The goal of this thesis is to determine if volumes and ejection fractions can be measured with more consistency and this is done by experimenting with imaging techniques using 4DViz software.

## 2. 4DViz

At Duke University, von Ramm Cardiac Ultrasound lab and associates decided to develop an ultrasound image processing freeware robust enough for researcher use. Cooper Moore programmed the initial 4DViz software which implements surface reconstruction models based on Chikai Ohazama's dissertation (Ohazama, 1998). 4DViz was written in C++ programming language and developed using Microsoft Foundation Classes. 4DViz is executable on Microsoft Visual Studios 2010. The software loads ultrasound volume data from DICOM files onto a display and renders three-dimensional models with Microsoft Direct3D application programming interface from DirectX 9. As freeware, 4DViz resources will not be tied to any company and can be used indefinitely.

## 2.1 Display Modes and Data

4DViz can display ultrasound volumes in 3D render and 2D slices. The graphical user interface on 4DViz has several available display modes that are similar to common displays found on many ultrasound machines:

- **Volume Render:** this mode displays a 3D model of the ultrasound volume data and allows the user to drag and click the model to change the view port. The user can also apply and adjust viewing cut planes.
- **Orthogonal Planes:** this mode provides two adjustable orthogonal cut planes for viewing the ultrasound volumes.
- **Slice View and Render View:** these modes are single slice cut planes through the volume data. The cut plane can be rotated and tilted with keyboard inputs to search the entire volume. Volume measurements, model making, and user inputs are done in Render View. Slice View will only display a single slice of any three-dimensional models created.

The graphical user interface also allows the user to adjust the data's contrast, gamma, brightness, and threshold.

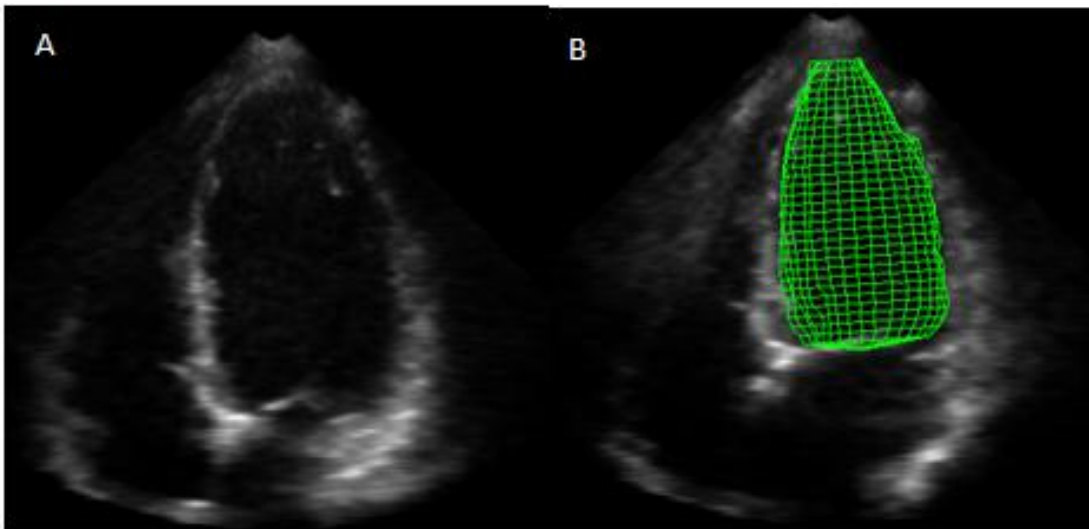
The DICOM files loaded by 4DViz are from Philips ultrasound machines provided by the Duke Clinic. The volume data has been scan converted into voxels aligned on Cartesian coordinate axes called Row, Column, and Depth. Depth is the axis from the top of the display to the bottom, while Row begins left to right. Each voxel is

assigned a value from 0 to 255 corresponding to display brightness. Brightness values of 0 are black and transparent in the display views. Brightness values of 255 are white and correspond to maximum reflected pulse pressures. Normally, most of the actual ultrasound image is a shade of gray ranging from 10 to 200.

The ultrasound images have had some processing from the Philips machines. The ultrasound images used in this thesis have all been processed with time-dependent gain to offset attenuation, so not all voxels farther in depth are dimmed. Because the volume data sets are videos of a beating heart, time is also a dimension used in the data. The volume data is sectioned into time frames during the cardiac cycle. The first frame occurs near end-diastole of the heartbeat. Subsequent frames lead to systole and the final frame leads back to end-diastole.

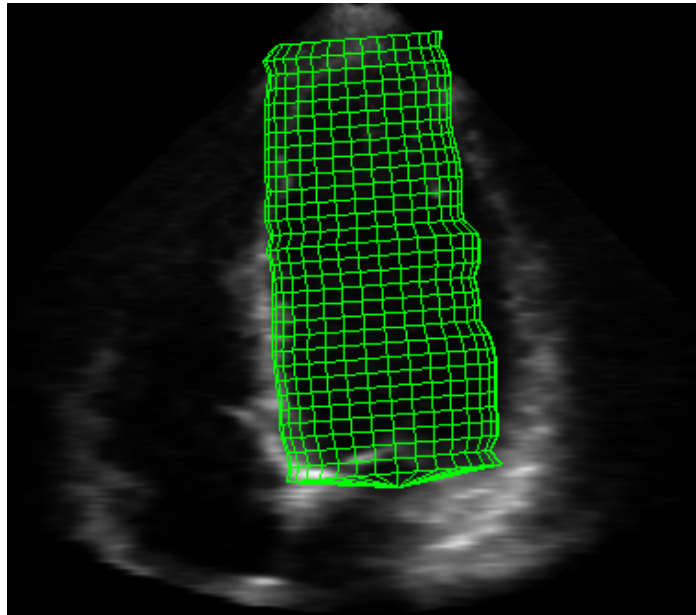
## 2.2 Surface Reconstruction Model

Chikai Ohazama designed a 3D surface reconstruction model that is built from user mouse clicks and interpolation. The model can be built on every frame in the data through 4DViz's Slice and Render modes. The 3D surface reconstruction model has an algorithm that calculates volumes using distances recorded in each DICOM file. To begin building the model, the user must click the display to place markers for ventricle apex and base, and then trace the ventricle wall from the ultrasound image. This must be repeated at several slice rotational angles to project an accurate model of the ventricle surface. Volumes generated are dependent on the user's mouse inputs and accuracy. Volume graphs are generated by modeling the surface on every frame of the ultrasound data. This is a time consuming task, taking several hours.



**Figure 3: (A) Render View of ultrasound volume data. A single slice of the left ventricle is shown. (B) The 3D surface reconstruction model applied over the ventricle on the systolic frame.**

The surface reconstruction model is conceptually cylindrical, a column of discs stored as radii at certain height and angle (theta). A weakness of this model is its inability to properly outline the planar structure of ventricle base, where the mitral valve is located. Only one radius can occupy a height and angle coordinate; a plane is impossible to store. This outlining flaw is shown in Figure 5. This issue at the base affects the thesis' algorithm and is addressed again in section 4.2.

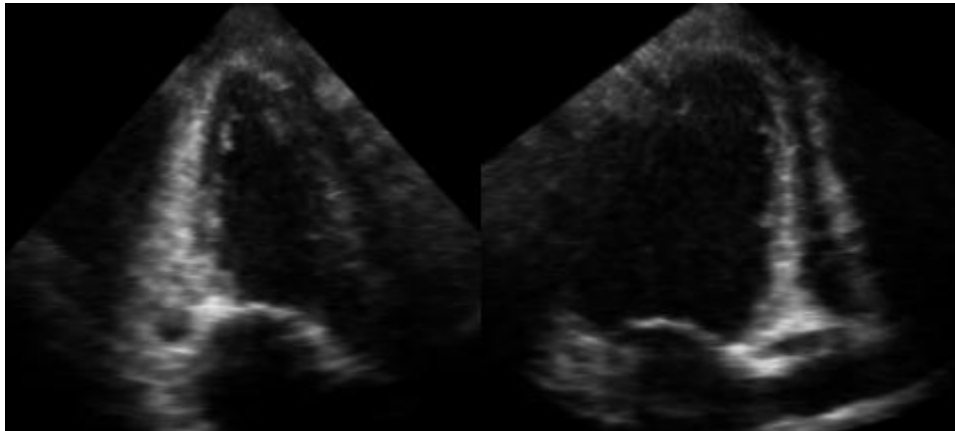


**Figure 4: The surface reconstruction model is cylindrical in design because the left ventricle can be modeled as a cylinder and volume can be calculated from cylinder equations. There is no way to model the planar base.**

The surface model does not interact with the original volume data. It is only displayed over the volume slice currently being viewed. But, the model can be translated into data points, and the thesis's algorithm does this to provide starting points for feature searching (discussed in section 4.2).

## 2.3 Initial Applications in Volume Charts

With the surface reconstruction model, 4DViz users can generate volume charts over a cardiac cycle. However, volume charts are difficult and results vary user to user. Accurately tracing the ventricle walls requires many rounds of practice. A complete 3D volume tracing in just one frame can take several minutes, and each volume data set can have a multiple frames. Furthermore, users must frequently estimate whenever the ventricle wall is not distinct in the image. Figure 5 below shows a pair of images for which tracing the ventricle wall is guesswork.



**Figure 5: In these ultrasound images, ventricle walls appear to be missing. This is fairly common due to the angular dependence of sound reflection. Tracing these missing ventricle walls can often be guesswork.**

Practice in tracing improves surface model quality and reduces the time involved. In a cardiac ultrasound course, eight students traced volume charts on the same ultrasound volume data. Their results are on the next page and used for testing in section 5.

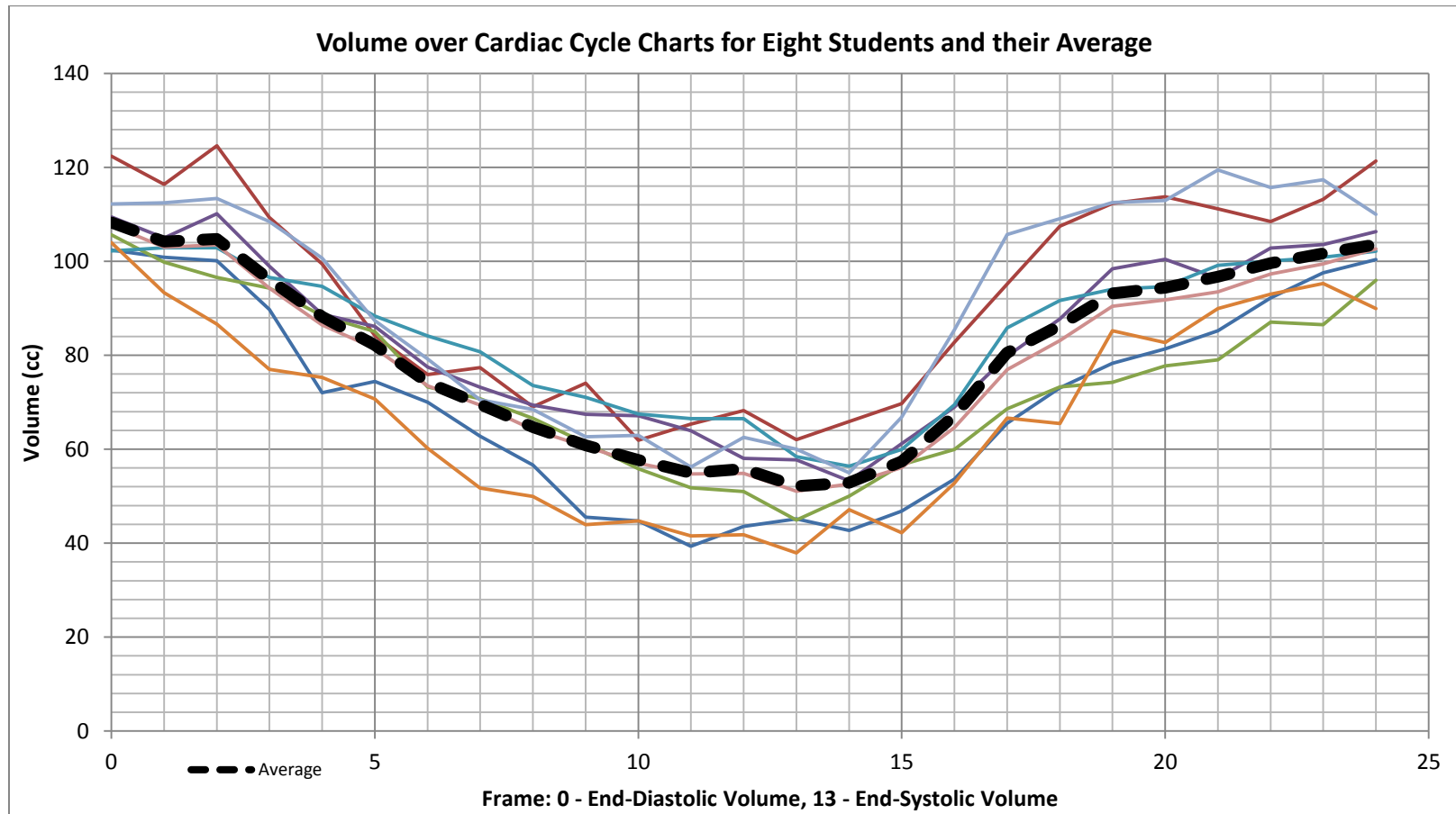


Figure 6: Volume charts measured by eight different students and their average overall. On many frames, a 20 milliliter range around the average shows the user dependence in producing consistent volumes on ultrasound images.

## 2.4 Hypothesis and Methods

This thesis intends to answer if 4DViz is robust enough for experimentation with ultrasound data. And, if so, what techniques can be developed to improve volume calculation consistency. The inconsistency in 4DViz surface model volume charts and the cardiologists' 10% ejection fraction error range both suggest that volume calculations are too dependent on examiner input. Variance is in part due to ultrasound's limitations, but ultrasound images do not change from examiner to examiner. Can an algorithm more consistently produce volumes using primarily the ultrasound data rather than examiner inputs?

Section 4 describes a new algorithm that can improve volume calculation consistency. Implemented in 4DViz, the algorithm reduces user input and incorporates knowledge of ultrasound data and heart anatomy. Accounting for ultrasound's own inconsistencies (random speckles, disappearing walls) is the largest hurdle. But with calibration, the algorithm's procedure shows interesting results. Calibration is done using water balloons of known volumes, imaged at the Duke Clinic.

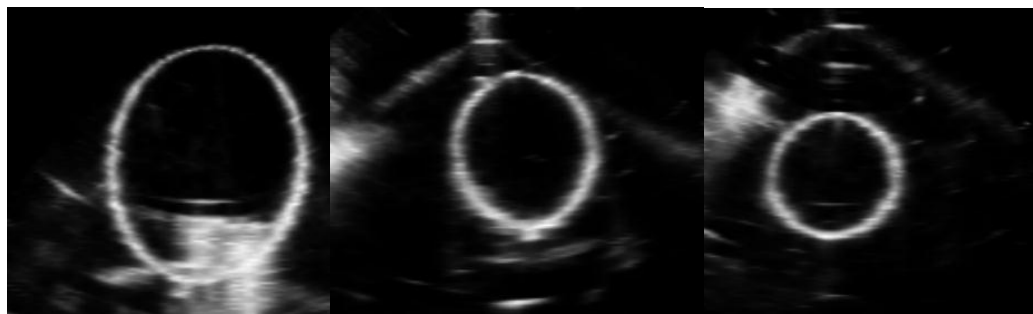
### 3. Applying Water Balloon Data and Heart Data

Images of water balloons with known volumes create a standard for calibrating algorithm parameters. Balloon images are ideal for tests because balloons have smooth surfaces and lack motion. Their ejection fractions should be zero. The algorithm's performance and errors can be measured by its volume results on the balloon images. Certainly, the left ventricle and a water balloon have different structure, but the balloons had volumes possible for the left ventricular and many calibrations operated well with the heart data.

There are no known volumes for any heart data set used in this thesis. Heart chamber volumes are not easily measured, but have been done with cadaver hearts that are dissected, filled, and then drained to acquire a known volume. This operation was not done for this thesis. Instead, the algorithm's results were compared to the class average of 4DViz volume charts (shown section 2.3).

### 3.1 Balloon Imaging

Three rubber balloons were filled with room temperature water and weighted on an electric scale to measure the volume. Water is an appropriate fluid as the speed of sound in room temperature water is roughly equal to the average speed of sound in soft tissue. The balloons contained 55.6 mL, 83.8 mL, and 110.0 mL volumes, all possible volumes of a normal left ventricle. Care was taken to remove any air bubbles contained in the balloon, as air produces image noise and causes submerged balloons to move. The filled balloons were submerged in a water tank and imaged with a Philips iE33 xMatrix Echocardiography machine, using an X5-1 3D transducer (Philips, 2012). The Philips machine was available at the Duke Clinic. Seven balloon volume data sets were captured: two of the 55.6 mL balloon, two of the 110.0 mL balloon, and three of the 83.8 mL balloon, which was handheld in place to mitigate motion. The other balloons were tied down to the tank with fishing wire, which is difficult to notice in the images.



**Figure 7: Ultrasound images of balloons. The leftmost has 110.0mL of water, the center has 83.8mL, and the rightmost has 55.6mL.**

## 3.2 Volume Data Sets of Hearts

The algorithm was applied to four ultrasound volume data sets to test for generality. All data sets are apical two-chamber views of the heart and the left ventricle is prominent in each image. A “normal” volume data set was used most often in testing because the imaged heart has no disease states. The other volume data sets were labeled: dilated, tilted, and myopathy. Dilated and myopathy data sets were expected to produce much lower ejection fractions than the normal data set. The tilted data set tested if the vertical position of the ventricle was a factor for volume calculations.

The Normal volume image is an incredibly high quality image taken by an expert sonographer. The ventricle surface is mostly distinct and noise from interference is low. The transducer was also well positioned to capture the entire ventricle upright through a complete cardiac cycle. So, the Normal volume data set is primarily used by participants in testing and for algorithm calibrations. The volume charts shown Figure 7 and used for comparing final results were measured from the Normal volume data set. The Normal data set is not immune to some of ultrasound’s challenges, however.

### 3.3 Challenges of the Ventricle Wall in Ultrasound

The dense ventricle wall is often visible on ultrasound images as bright borders. But, the ventricle wall is laced with noise that can mislead the algorithm. Ultrasonic reflection is angularly sensitive and the ventricle wall is not a smooth surface; it is a bumpy surface with several round muscular columns called trabeculae aligned from apex to valve. Trabeculae are not easily distinguishable in an image (see Figure 8), but do manifest as seemingly random features within the ventricle chamber. This feature randomness, combined with possible interference noise or reverberation artifacts, are detrimental to image processing techniques that rely on knowing bright feature locations. Furthermore, features can disappear and reappear due to the motions of the cardiac cycle. For image processing, the wall can be difficult to use.



**Figure 8: The interior wall of the left ventricle lined with trabeculae. The circle highlights possible noise from trabeculae.**

## 4. The Filler Algorithm Techniques

This section explains an algorithm that identifies and counts voxels filling the ventricle walls on every frame of the data set. When multiplied by the volume per voxel supplied in each DICOM file, the sum of the identified voxels equals the volume of the ventricle chamber. Accurately identifying the chamber voxels is accomplished through a series of programming steps: applying thresholds, identifying features along the ventricle wall on an initial frame, tracking features through all frames, and constructing a surface from those features. Only after a surface is constructed can the algorithm begin the last step of identifying voxels, by essentially “filling” the surface. This section will discuss each step’s importance, execution, and calibration.

The initial step of identifying features on a frame is especially important and is where the algorithm attempts to reduce user input with two methods. The first method requires the user to create a surface reconstruction model on just one frame instead of every frame. The second method only requires the user to click twice on one frame to identify two key ventricle features: the base and apex. The subsequent programming steps are all the same, but the user is less involved in creating the initial features for the algorithm to track. Comparison of the methods is done in section 5.

## 4.1 Applying a Threshold to the Data

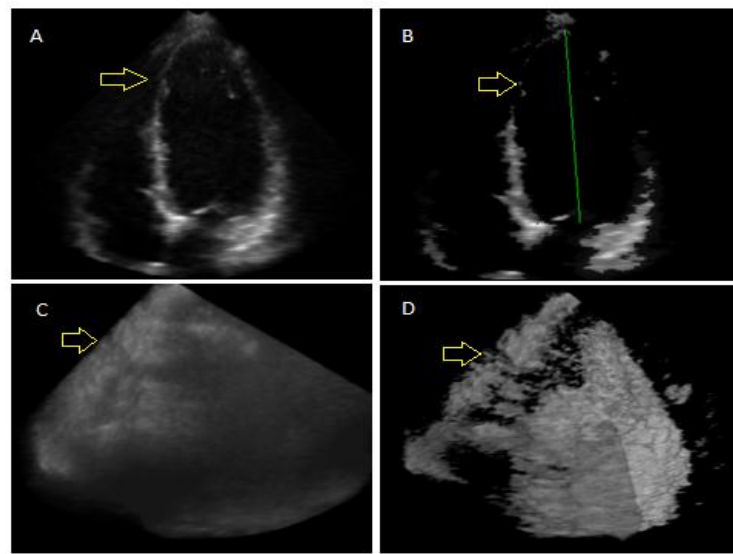
The initial step is applying a brightness threshold to the data. While not necessary for the algorithm, applying a threshold to the data reduces runtimes and can simplify following steps. Parts of the algorithm may not need every voxel. Instead the algorithm can ignore any voxels below the brightness threshold. A large fraction of the data is unimportant to the algorithm because the features needed later are typically highest in brightness, ranging from 80 to 200. It's unlikely that an important feature such as the ventricle wall or mitral valve will have low brightness.

Threshold ultrasound brightness requires regional consideration. Ultrasound images are not uniform; one threshold applied to the entire image is not effective. Voxels closer to the transducer are inherently brighter, and the ventricle is only in the top region of most images. Regional thresholds should be more effective. To test this, the algorithm divides the ventricle data into eight equal octants and then, for each, applies a different threshold.

Choosing an appropriate threshold for each octant was initially done by generating a regional histogram and binning all non-zero brightness. In a histogram, threshold values can correspond to fractions. For example, the 10% brightest voxels in an octant can correspond to a threshold of 60. Choosing to keep the brightest 10% of voxels in the Normal data set successfully highlights the ventricle walls in some regions, but not all. In regions close to the transducer, bright voxels in the epicardium skewed

the histograms. Thresholds were too high, removing dimmer portions of the ventricle wall. Figure 9 on the next page shows this problem. Regional histogram thresholding problematically removes the ventricle surface.

A more precise thresholding technique is necessary. A spherical search from the center of the ventricle seeks out the first voxel to pass an increasing threshold. The threshold is iteratively incremented as the spherical search repeats until a maximum number of voxels exceeding threshold is found in a target region around the ventricle surface. The region applies knowledge of left ventricular dimensions and avoids the problem of extraneous brightness. This thresholding method is useful for the algorithm's method of automatic feature searching.



**Figure 9: (A) and (C) are unaltered ultrasound volumes in Render and Volume view modes, while (B) and (D) are the same but have had a regional threshold applied. In (B), the ventricle wall at the yellow arrow is gone because the threshold was skewed. Bright voxels pointed to by yellow arrows in (C) and (D) skewed the histogram.**

## 4.2 Feature Searching with the Surface Model

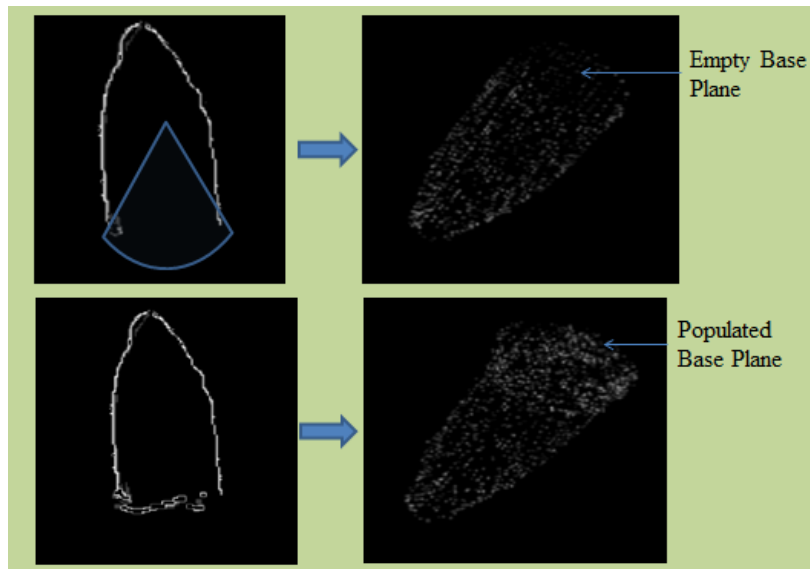
Speckle is a scatterer and interference based phenomenon in ultrasound imaging. Depending on its usefulness, speckle has also been called “features”, “noise”, or “patterns”. Speckle appears on images as brightness with the size of single resolution cell, determined by depth, pulse width, frequency, and transducer size. The ventricle wall is a dense layer of scatterers, and speckles at wall interfaces form features. In this thesis, features are treated as 3x3x3 voxel cubes and their movement from frame to frame can be analyzed accurately as tissue motion (Bashford, 1995). Scatterer motion corresponds to feature motion.

But which features are the best to track? In order to track, initial condition features on the ventricle wall must first be identified on a single frame. Noise, interference, or reverberations can produce features as bright as from the ventricle wall, but their features are momentary due to motion and are filtered when tracked. User judgment on selecting the initial condition features simplifies the algorithm at the cost of introducing user dependence. The algorithm attempts two different feature searching methods that reduce user dependence but increase algorithm complexity.

One method applies Ohazama’s surface reconstruction model. With the surface reconstruction model, the search algorithm for feature candidates is relatively straightforward if the user can trace an accurate model on one frame. The 4DViz user estimates when no wall exists, so the model can reasonably account for that ultrasound

challenge. The algorithm translates the model's overlay into the Row, Column, and Depth coordinates used to index the data voxels. Model points become the centers of 3x3x3 cubic searches for local maxima, which are identified as features for simplicity.

The base has important features to track because it has rapid motion. Section 2.2 explained that the surface reconstruction model does not account for the base. To compensate, the algorithm automatically searches a downward cone in the ventricle chamber for base features (shown in figure below). The automatic detection method for base features is described in the next section and leads to the algorithm's second method: automated feature searching through a sphere instead of a cone.

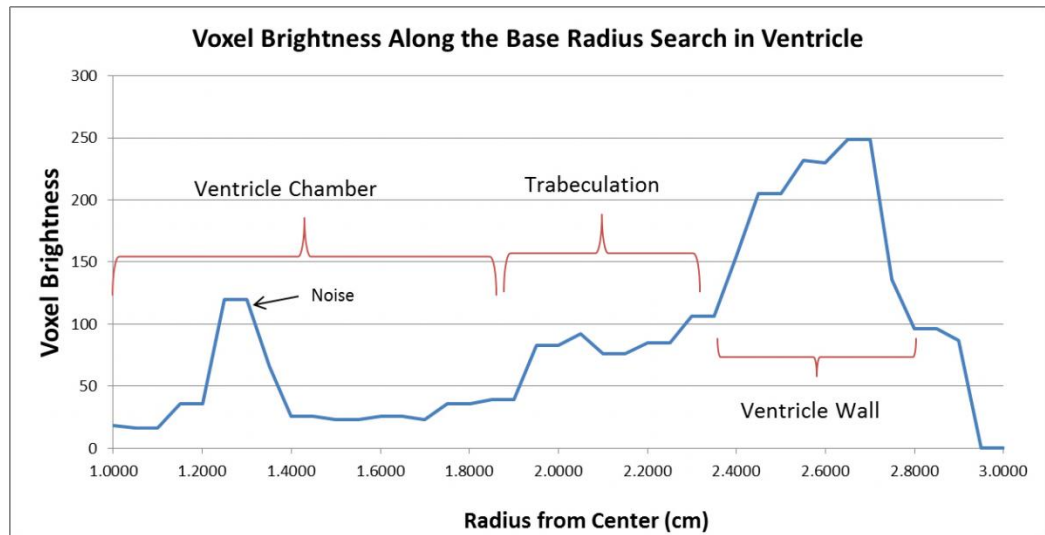


**Figure 10: Feature searching with and without base features. The translated surface model is shown in 2D on the left and 3D on the right (tilted with base pointing upward). The introduction of base features is shown on the bottom half.**

### 4.3 Automated Feature Searching

The method of feature searching in section 4.2 allows volume calculation from one user traced surface reconstruction model, effectively dividing the time needed to create volume charts by the number of frames in data. Another method attempts to reduce this time further with near-automatic volume calculations. In this second method, the user only identifies the ventricle apex and base, a task that takes a few seconds rather than several minutes of tracing. Automatic volume calculations should provide the most consistent volume measurement but the algorithm is more complex. Accurately locating initial features along the ventricle wall and not before or after, can be complicated for a program.

The automated method begins with a spherical search from the center of the ventricle, approximated by the user's mouse clicks at the ventricle apex and base. The center will not be at exactly the same location in ventricle space from user to user, and this will give minimal variance to volumes. But, the automated search functions as long as the center is within ventricle space. The spherical search is comprised of 1x1 degree pyramids that search at every spherical angle and at every half millimeter along a radius range of one centimeter to half the apex-to-base distance (the longest dimension of the ventricle). All voxel brightness values that intercept the pyramid radius are stored by their distance to the center. Brightness charts are created with every pyramid and are evaluated to identify possible feature locations. An example is on the next figure.



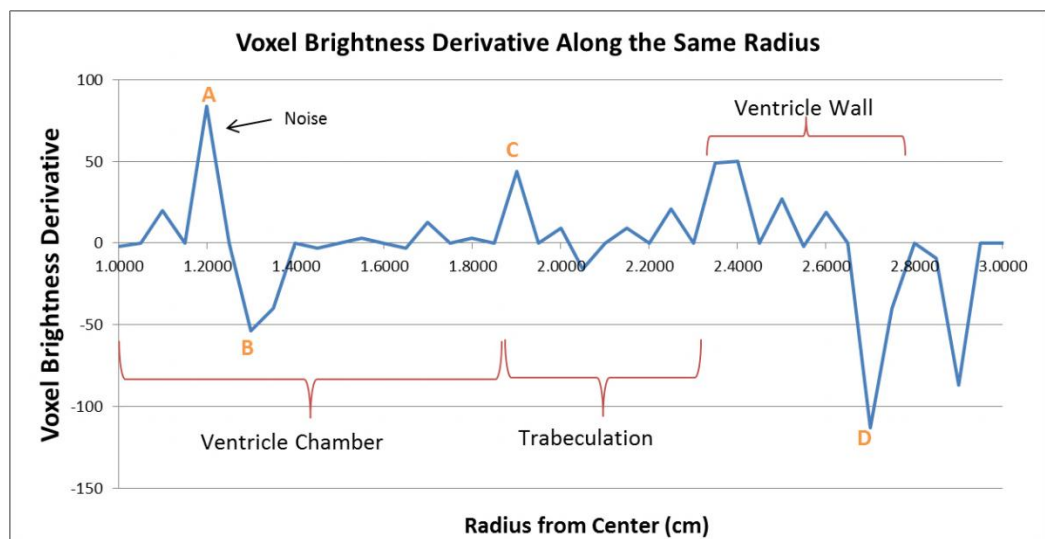
**Figure 11: A brightness chart of all voxels along the first search radius. The search range is just 1cm to 3cm in the Normal Heart data set. Ventricle chamber, trabeculation, and ventricle wall can be identified visually.**

Many of the brightness charts have three identifiable sections: leftmost dim ventricle chamber voxels, rightmost bright ventricle wall voxels, and a medium bright range at the center. This medium bright range is trabecular reflection near the ventricle wall; because of its columnar structure, trabeculae do not reflect ultrasonic pulses as well as flat surfaces. The ideal location to track features would be at the interface of the trabeculation and the ventricle wall.

In many cases, this interface is not trivially found by the algorithm.

Trabeculation onset can be identified and then offset, but spurious noise in the ventricle chamber can mislead this technique. For example, if the first value above a threshold is considered the prime feature location, then in Figure 9's case, the technique would erroneously place the feature right at the random noise. A more accommodating

technique developed from an observation made on the shape of the trabeculation and wall interface: there is a rise in brightness, followed by a long plateau, and then another higher rise in brightness. The long plateau contrasts with the noise's short burst of brightness. Now dealing with rises and falls, the algorithm creates derivative charts (dBrightness over dRadius) from the brightness charts.



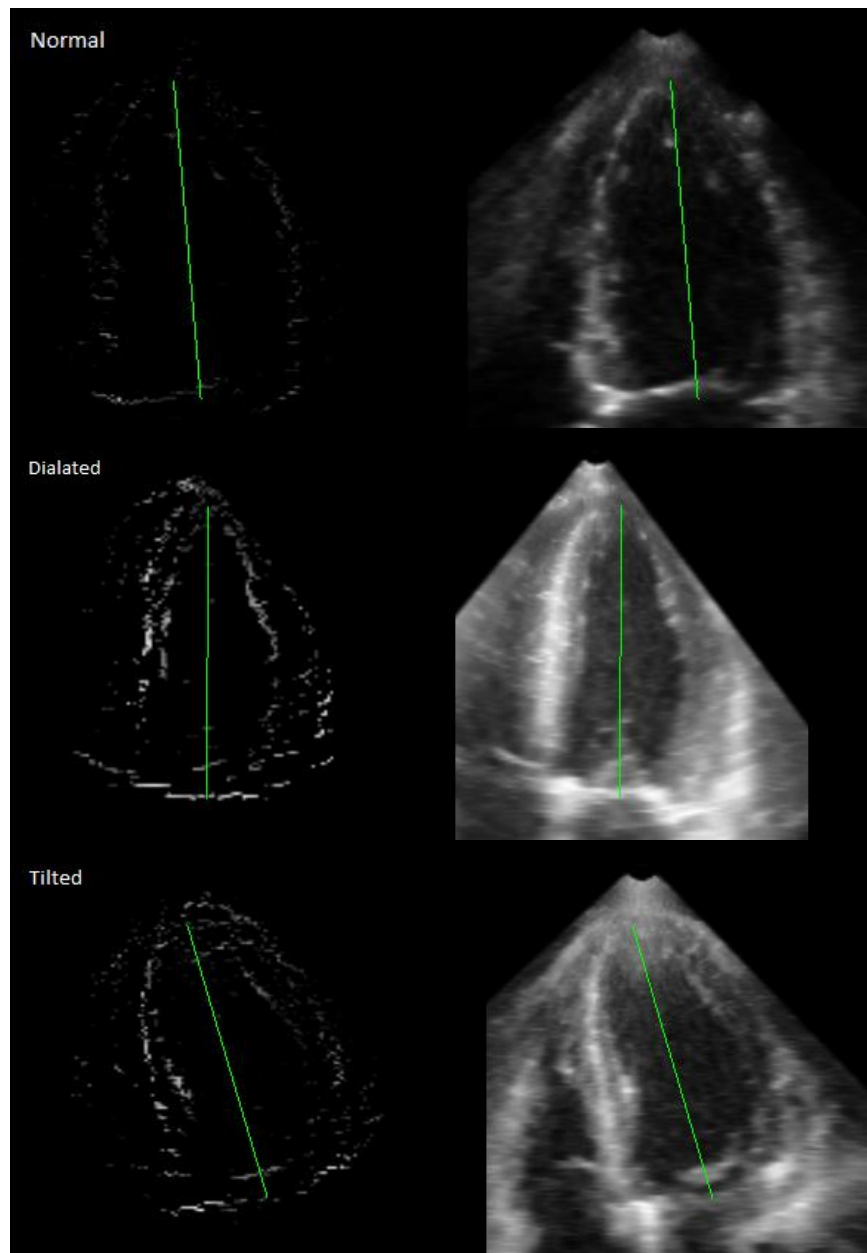
**Figure 12: A derivative chart of Figure 11. The noise is easily spotted with its large rise (A) and steep fall shortly after (B). Recording the distance between a significant rise to a significant fall can identify where trabeculation and ventricle wall are located, in this case between (C) and (D).**

The derivative chart allows a high-low pairing program to identify both noises and wall features. Trabeculation and walls are thick, roughly a centimeter of endocardium, and any noise is likely to be short. The program travels along the radius axis until it records a positive derivative of greater than 20. The program then continues along the radius, seeking a negative derivative of less than -20. The distances between positive to negative peaks are measured. The longest distance is selected as the most

likely to be trabeculation and wall. Since trabeculation is roughly 4 millimeters in length, the program applies an offset to the trabeculation onset and locates the interface.

The derivative program was calibrated several times by changing thresholds, radius ranges, pyramid sizes, and derivative thresholds. In order to make automatic feature search generally applicable to all volume sets, it was often tested on the four available image sets. Some results are shown in the next figure. With calibration, the automatic method became less sensitive to noise and does nearly outline the ventricle surface. However, it also draws many inaccurate clusters, usually in the bright epicardium and mitral valve structure. Feature searching in those clusters would produce many errors.

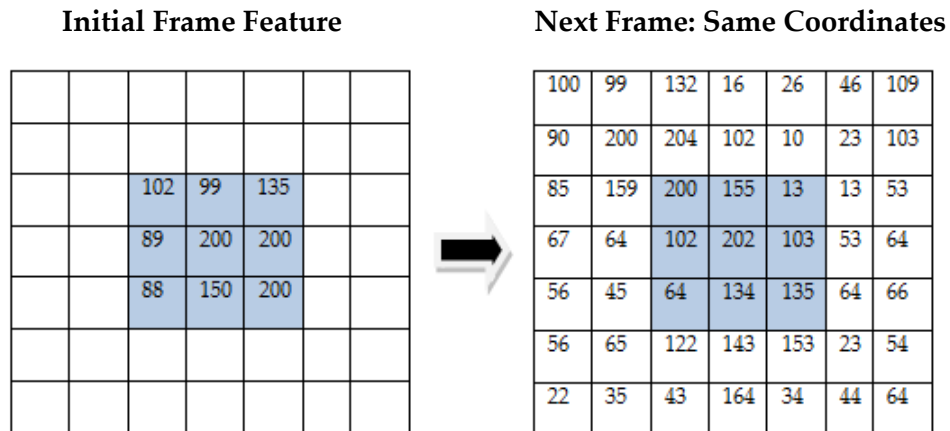
Errors in feature searching can drastically alter the results of the remaining algorithm steps. In particular, feature tracking compounds the errors over several frames. Compensating for the inaccurate clusters, a density filter can keep only features search locations that are adjacent to at least two other feature search locations. This removes many of the incorrect features, as well as some correct features, but somewhat improves volume calculations.



**Figure 13: The left images show the results of the automatic feature search and the right images show the original ultrasound image. A ventricle is identifiable in the automatic results, but many features are at incorrect locations as well.**

## 4.4 Sum of Absolute Difference Tracking

The remaining algorithm steps are the same for both methods of feature searching. Once features are identified, the next step is to track their position through all frames. The algorithm accomplishes feature tracking and motion estimation with a proven sum of absolute difference method (Geiser, 1999). When local maxima voxels are identified, they are expanded to 3x3x3 cubes because features in the image may be larger than a single voxel. The cubes contain brightness values in initial frame are compared to the next frame's 3x3x3 cube at the same Row, Column, and Depth coordinate. The initial 27 voxel brightness values are subtracted from the next frame's 27 voxel values and the absolute values of the differences are summed to create a tracking metric. A two-dimensional representation of this procedure is shown in the figure below:

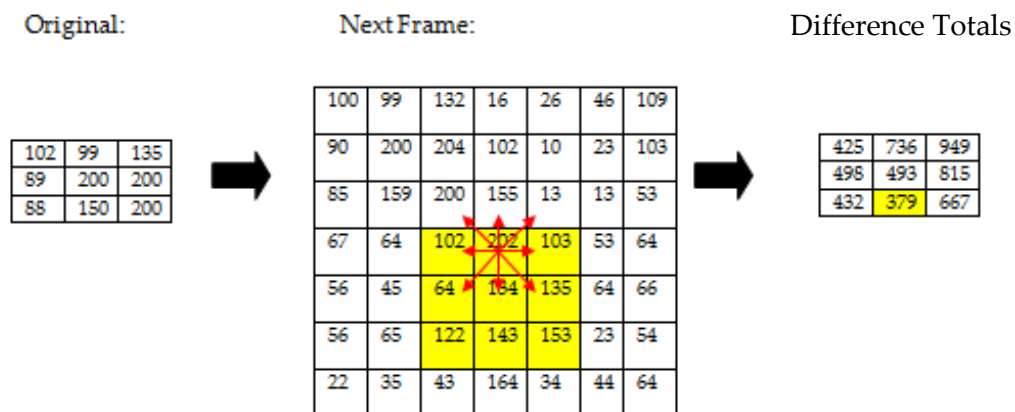


**Figure 14: Comparing a 3x3 square to another 3x3 square in the next frame. The sum of the absolute difference of the squares is 493.**

Sum of absolute difference (SAD) is the metric comparing the likelihood a next frame's feature is the previous frame's feature. A smaller total is a greater likelihood because the differences from one frame to the previous are minimal. If setting a SAD threshold, a stingy value is beneficial because if a feature cannot track to something nearly exactly like it somewhere in the next frame, then it was likely not an appropriate feature to track at all.

Since the feature tracking is motion tracking, features can move their coordinates in the next frame. It is appropriate to calculate a SAD from a range of voxels around the original coordinate. The range must be incremented until the sum threshold is met or the range increases to distances that not physiologically possible. It is not likely that features can travel ten voxels (about 1 cm) in fractions of a second. If the search range grows too large, the feature is discarded and the algorithm continues with the next feature. Figure 15 demonstrates this expanding range procedure on the next page.

A. Applying a difference threshold of 400



B. Expanding the search and applying a new difference threshold of 100

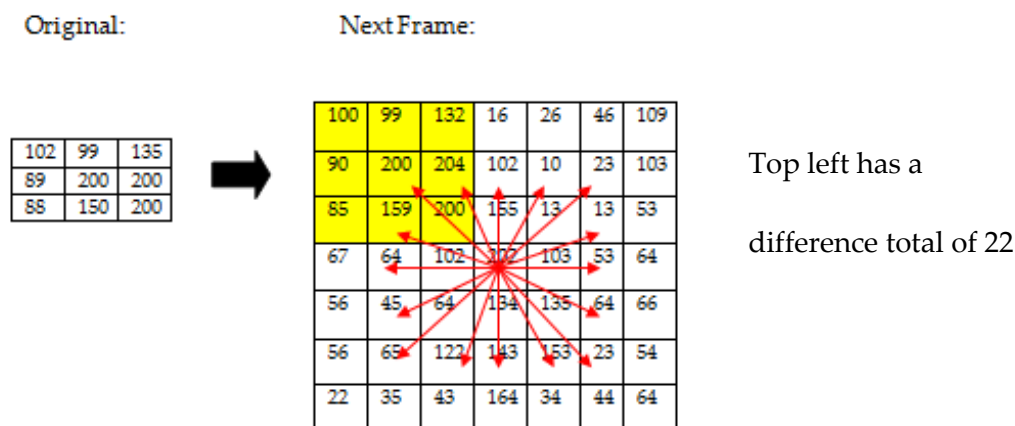


Figure 15: An example of why low correlation thresholds are beneficial. (A) A sum of absolute difference threshold of 400 will track the feature down one pixel with a difference total that barely passes the threshold. (B) No difference total from (A) meets the new threshold of 100 so the search range is expanded in red arrows. The top left corner of the image is the optimal location to track the original feature.

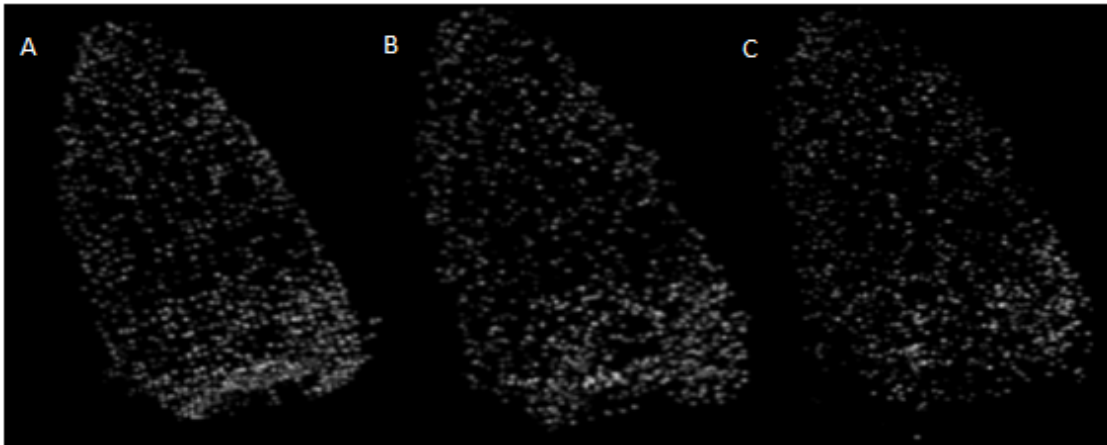
Many errors occur if features are not tracked correctly, tracked too tightly, or not tracked tightly enough and these errors compound over frames. If the maximum search range for taking sums of absolute difference is too large, wall features can track to bright regions outside of the ventricle wall. However, if the search range is too small, the rapid motion of the base cannot be tracked. Several techniques improve feature tracking in the algorithm.

One technique is to select the end-systolic frame as the initial frame. The end-systolic frame's ventricle walls will have brighter overall features because the ventricle has condensed and gotten closer to the transducer at systole. Bright features are more likely to remain bright through all frames. Also, because the end-systolic frame is normally in the middle of the frames, it is possible to track features forwards and backwards. Though done twice, the frame distance in tracking is halved, so errors compounding over frames are reduced. All future tracking applications of 4DViz should consider beginning at the end-systolic frame.

As mentioned previously, the base's rapid motion must be tracked successfully. Using increased cube sizes for the base features can improve base tracking. The algorithm identifies base features by selecting a number of features with the lowest depth coordinates. These chosen base features are expanded to a 5x5x5 cube instead of the 3x3x3 cubes. 5x5x5 cubes are more difficult to track by mistake, as the sum of absolute difference accounts for 125 voxels. Picky base features will search farther each

frame, corresponding to rapid base motions. The only tradeoff for this larger base feature tracking is increased algorithm runtimes.

Once feature tracking has completed in a frame, the feature coordinates are recorded into a dummy volume called Selected Features. Selected Features can be viewed in 4DViz Render Mode and is a visual way to compare results. The tracked local maximum voxels in Selected Features are highlighted by making their brightness values equal to the maximum 255. Selected Features can also be viewed in Volume mode as seen in the figure below. The next section discusses how SAD thresholds and search ranges were chosen.



**Figure 16: The dummy volume, Selected Features, records local maxima from the tracking algorithm. (A) is the initial end-systolic frame. (B) is 2 frames later. (C) is 6 frames later. The organization of the features spreads out over time.**

## 4.5 Calibrating SAD Threshold and Search Range

Finding an optimal SAD range can improve the feature tracking processing step. SAD range is tied to the SAD threshold. If the threshold is low, the average range increases because features search farther in voxels to meet tougher thresholds. This can be seen in Table 1 on the next page. The SAD algorithm was iterated several times on the 110mL balloon image. On each iteration, the SAD threshold decreased by 1 (from 9 to 1) and forced more accurate matches for feature tracking.


Two histograms were generated, recording the SAD search range and SAD total for every successful feature tracked over all 19 frames of the 110mL balloon image. The first histogram shows that as SAD threshold decreases, the search range for features generally increases to find passable SAD totals. However, the effect is not significant until the threshold is 1 or 0; the large majority of features only need to search within range of 1 voxel to track. This is understandable because the balloon is mostly motionless, so features should not be moving.

The second histogram recorded SAD totals. As SAD threshold decreases, features with higher totals are forced to find lower totals by expanding their search range. The percentage on the right side of the table reveals how many features during the entire tracking step must increase their search range to successfully track. 98% of balloon features are achieving SAD totals of 3 or lower.

**Table 1: Two 3x3x3 SAD Histograms for 110mL balloon Showing the Effect of Thresholding**

For Balloon  
110 mL

SAD Threshold	9	8	7	6	5	4	3	2	1	0
Search Range										
1	23096	23094	23084	23070	23023	22864	22637	21846	19664	11831
2	4	6	15	28	73	194	430	1085	2794	6110
3	0	0	1	1	2	4	17	119	483	3242
4	0	0	0	0	0	2	2	7	74	1295
5	0	0	0	0	1	1	1	0	6	362
6	0	0	0	0	0	0	1	1	5	84
7	0	0	0	0	0	0	0	1	1	29
8	0	0	0	0	0	0	0	0	1	13
9	0	0	0	0	0	0	1	1	0	8



SAD totals per  
Feature

Feature	9	8	7	6	5	4	3	2	1	0	% of total
(perfect match) 0	11474	11475	11473	11481	11500	11544	11662	12125	13529	22967	49.7
1	7978	7978	7989	7993	8011	8049	8184	8489	9499		34.5
2	2365	2365	2365	2364	2372	2373	2386	2446			10.2
3	798	799	799	801	803	812	857		0		3.5
4	291	291	291	292	289	287		0	0		1.3
5	124	124	123	122	124		0	0	0		0.5
6	45	45	46	46		0	0	0	0		0.2
7	14	14	14		0	0	0	0	0		0.1
8	9	9		0	0	0	0	0	0		0.0
9	2		0	0	0	0	0	0	0		0.0
10		0	0	0	0	0	0	0	0		
	23100	23100	23100	23099	23099	23065	23089	23060	23028	22967	

From this data and 4DViz observation, setting the algorithm's SAD threshold to 3 would be optimal. The low threshold forces close matches and 2% of the features expand their search ranges. Volume calculations on the 110 mL balloon using a SAD

threshold of 3 yielded only a 5% difference from largest volume to smallest. The SAD threshold of 3 was then applied to the Normal Heart data set and the ejection fractions produced by surface reconstruction model were a close match to the class average ejection fraction: 48.4% to class average 48.2%. When viewing the location of features relative to the ventricle wall on several frames, it was apparent that most features were tracking near the ventricle walls.

## 4.6 Approximating the Ventricle Surface with Lines

Once features have been tracked on every frame, the next processing step is to generate a continuous surface that can approximate the ventricle wall. The algorithm's ideal surface has a thin porous interior without many large gaps. A porous interior can simulate the trabeculae layer in the heart. Because of trabeculae, the ventricular surface is bumpy and unpredictable; the columns of myocardium slightly reduce the blood volume in the ventricle. The algorithm approximates this reduction by creating a porous, cobweb-like surface of lines connecting many features.

Drawing lines that connect features can be done with the following reasoning. For one voxel to connect to another in volume space there are only 26 possible surrounding voxels to travel to in a Cartesian coordinate space. The correct voxel to draw in a line is the voxel that has the shortest distance from both original voxel and target voxel and can never be the previous voxel. Iterating this reason draws a line connecting two voxels. This routine is done in the function *AddLine* in 4DViz.

Using *AddLine*, each voxel in Selected Features draws to all other feature voxels in an adjustable range. If the range is too small, then the constructed surface will have too many gaps to effectively use *Filler*, the next and final step in the algorithm. If the range is too large, then a different problem occurs: the surface becomes too thick in areas of high feature concentration such as the apex. The apex's rounded point concentrates many features in a small region and lines begin drawing into the ventricle chamber. This

effect can be seen in Figure 17 (C). Calibrating the line drawing range on water balloons is discussed in the next section.

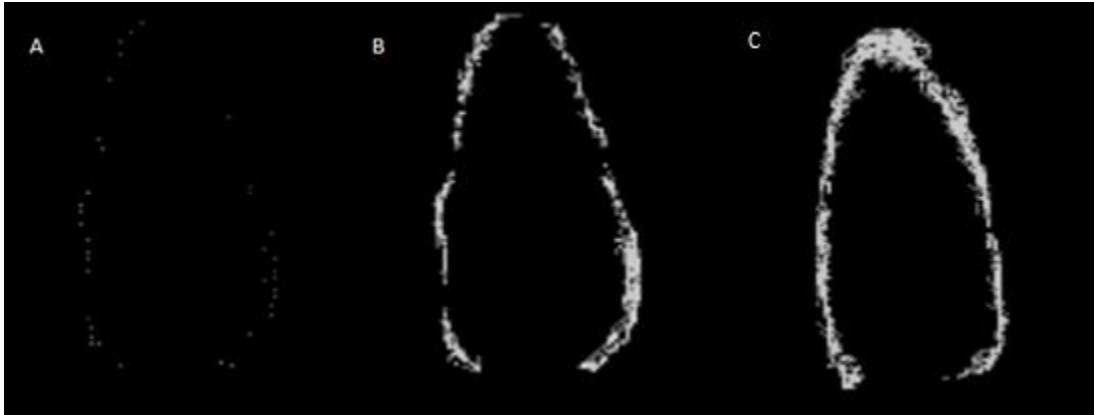


Figure 17: (A) shows the local maxima voxels on a single cut plane in Select Features. (B) shows the algorithm's use of *AddLine*. Notice that there are some gaps in the surface, but not too many and none too large. The base has no lines because the features are too far apart. In (C) the search range is doubled and the apex is now too thick.

## 4.7 Calibrating Line Drawing Range

The *AddLine* range of voxel to voxel lines was calibrated to produce the most gap-free consistent surface and to avoid drawing lines into the ventricle chamber. Because gaps reduce volumes and lines in the chamber reduce volumes, the optimal *AddLine* range produces peak volumes after the voxel filling step. Varying *AddLine* range in algorithm tests of different images revealed that range values of 9 to 11 voxels acquired peak volumes, but that these ranges were not consistent between images. Ventricle size and shape may be the reason.

It was hypothesized that smaller ventricles and balloons with compacted features will require smaller line searching range to prevent drawing lines in ventricle chambers. The algorithm was applied to the 110 mL and 55.6 mL balloon images to verify the importance of volume size in *AddLine*. Tables 2 and 3 record the calculated volumes on the first, last, and initial frames. A histogram of the number of lines drawn by each feature over all frames was generated to find trends. Repeating the algorithm several times and changing only the line drawing range did not confirm the hypothesis. Both the 55.6mL balloon and 110.0mL balloon had peak volumes on all frames when the search range was 9 voxels. At that search range, most features on the 110.0mL balloon connected to 11 to 19 other features. The 55.6mL balloon had similarly connected to 12 to 20 other features. Based on these results, the *AddLine* search range was set to 9 and then tested on the Normal heart. The results of the Normal heart test are on Table 4.

**Table 2: Results of AddLine Tests on 110.0mL Balloon**

Search Range	5	6	7	8	9	10	11
Filler Volume (ml)	60.41	105.7	117.75	121.65	124.2	123.5	122.4
% of actual volume (110.0)	55%	96%	107%	111%	113%	112%	111%
First Frame Volume (ml)	53.9	93.2	115.6	123.7	124.7	123.5	121.4
Last Frame Volume (ml)	53.5	92.9	116.4	124	124.7	124.5	122.7
# of Lines drawn per Feature							
0	0	0	0	0	0	0	0
1	47	8	2	1	0	0	0
2	223	21	6	5	4	3	0
3	831	65	9	2	2	1	2
4	1966	258	29	5	0	2	0
5	3257	676	71	9	2	0	2
6	3955	1360	205	30	1	0	0
7	3779	2284	513	77	11	5	3
8	3190	3036	895	185	25	3	1
9	2120	3320	1506	326	52	14	5
10	1183	3078	2094	637	106	24	6
11	577	2400	2479	1034	214	27	7
12	241	1815	2692	1458	422	74	17
13	109	1175	2443	1886	623	133	27
14	28	584	2182	2162	1002	245	54
15	5	303	1539	2261	1342	428	92
16	2	149	1123	2088	1618	644	171
17	0	54	722	1714	1893	867	276
18	0	35	411	1405	1901	1196	422
19	0	8	229	1069	1721	1356	604
20	0	3	114	728	1493	1567	836
21	0	0	59	479	1270	1583	996
22	0	0	38	254	988	1500	1186
23	0	0	16	185	795	1298	1298
24	0	0	2	92	535	1136	1360
25	0	0	3	68	327	907	1223
26	0	0	1	33	212	736	1178

\* highlight is maximum

**Table 3: Results of AddLine search range tests on 55.6mL Balloon**

Search Range	5	6	7	8	9	10	11	12
Filler Volume (ml)	25.03	38.26	45.15	46.45	49.98	49.86	49.83	48.81
% of actual volume (55.6)	45%	69%	81%	84%	90%	90%	90%	88%
First Frame Volume (ml)	15.78	32.3	43.7	49.6	51.7	51.3	49.9	48.5
Last Frame Volume (ml)	17.1	31.4	43.5	49.8	50.7	50.1	49.7	48
# of Lines per feature								
0	0	0	0	0	0	0	0	0
1	9	2	0	0	0	0	0	0
2	39	2	1	0	0	0	0	0
3	245	29	1	0	0	0	0	0
4	529	93	10	0	0	0	0	0
5	918	270	26	4	0	0	0	0
6	1039	500	97	9	0	0	0	0
7	983	720	213	26	5	0	0	0
8	792	821	388	83	10	0	0	0
9	490	867	553	162	29	5	0	0
10	291	677	656	298	57	9	1	0
11	116	516	721	416	121	25	5	1
12	64	376	590	565	216	57	11	2
13	23	199	523	540	335	88	23	9
14	8	131	414	546	399	163	52	10
15	2	56	262	534	517	223	84	19
16	0	34	191	398	480	297	117	47
17	0	7	136	341	456	406	184	64
18	0	1	64	218	435	388	232	103
19	0	1	40	168	324	409	280	134
20	0	0	22	104	264	370	314	178
21	0	0	9	72	186	363	351	226
22	0	0	3	37	149	286	331	284
23	0	0	0	30	93	227	329	275
24	0	0	0	7	59	176	297	283
25	0	0	0	4	43	128	262	276
26	0	0	0	1	19	88	185	296

\*highlight is maximum

**Table 4: Results of AddLine search range tests on Normal Heart Image**

Search Range	5	6	7	8	9	10	11	12	13
Filler Volume (ml)	18.56	36.8	50.77	55.76	60.65	63.53	63.97	63.73	63.3
% of actual volume (52.15)	36%	70%	97%	107%	116%	122%	123%	122%	121%
First Frame Volume (ml)	6.93	18.1	36.6	54.7	71.1	83.1	90.2	93.4	94.9
Last Frame Volume (ml)	7.71	21.4	39.4	59.9	76.2	87.4	92	93.6	93.8
# of Lines per feature									
0	0	0	0	0	0	0	0	0	0
1	675	231	94	44	23	6	4	0	0
2	1502	489	189	82	45	32	18	12	1
3	2473	925	313	126	59	37	27	16	14
4	3160	1476	532	187	81	43	23	15	11
5	3301	2113	817	293	107	59	27	18	11
6	2894	2525	1318	489	199	86	58	34	16
7	2145	2543	1623	763	242	100	57	42	33
8	1362	2414	1959	1071	462	156	63	36	21
9	740	2025	2092	1350	633	230	77	39	27
10	347	1477	2104	1564	866	355	143	56	30
11	165	939	1845	1700	1037	500	206	71	49
12	56	600	1559	1785	1309	719	276	111	59
13	14	377	1223	1744	1375	871	373	195	69
14	5	187	856	1596	1473	987	556	232	110
15	1	75	593	1327	1549	1101	691	343	131
16	0	34	348	998	1556	1195	773	378	204
17	0	11	215	743	1297	1262	980	502	248
18	0	4	123	536	1107	1292	1024	648	316
19	0	1	56	331	895	1305	1105	766	429
20	0	0	29	232	680	1173	1144	921	487
21	0	0	20	121	504	978	1130	901	567
22	0	0	5	71	355	796	1075	981	695
23	0	0	1	61	256	709	974	982	789
24	0	0	0	36	157	534	857	973	850
25	0	0	0	10	119	381	742	921	904
26	0	0	0	2	69	301	588	817	900

\*highlight is maximum

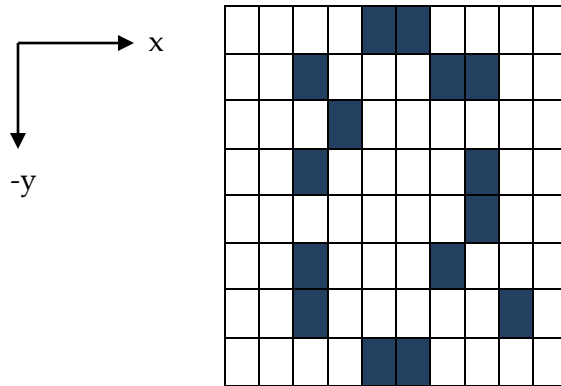
*AddLine* range of 9 did not generate the peak volume in the Normal Heart data; *AddLine* range of 11 was superior, especially for the first and last frames. The Normal Heart histogram showed that lines drawn per feature were more disperse. There are two possible explanations. First, the balloons' spherical shape does not represent the ellipsoidal ventricle well enough. Second, the motion of the heart spreads all features and so a larger range is required to draw a gap-free surface. This is evidenced by the first and last frame volumes which increase as search range increases even while the initial frame's volume decreases.

A suggestion for future work would be to implement an *AddLine* search range that increases over frames. This should account for some of the feature diffusing. For all remaining heart tests on volume, the search range was set to 11 voxels.

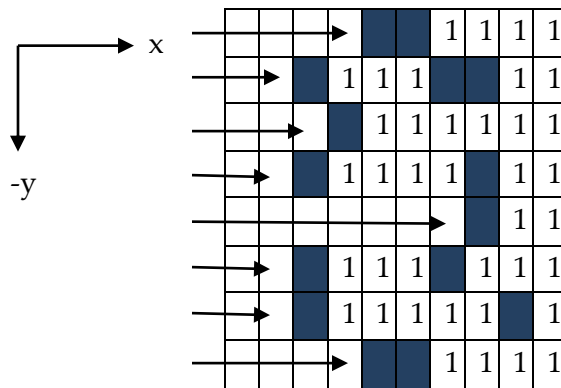
## 4.8 The Filler Function

The final step in the algorithm is determining which voxels are inside the ventricle's walls and summing those voxels. Assuming *AddLine* generated a reliable surface with few gaps in the Selected Features volume space, the function *Filler* should operate by sequentially marking voxels if they are behind a surface from any one of six perspectives in 3D Cartesian. The six perspectives are aligned to directions of positive and negative Row, positive and negative Column, and positive and negative Depth. From each of these directions, the *AddLine* surface casts a "shadow" on the ventricle chamber. A "shadow" is abstracted by searching along a perspective until the first non-zero value is reached and then incrementing every voxel value afterwards by one. Because the Selected Features volumes start as a collection of bright (>200) voxels and zero voxels, incrementing by one essentially marks a voxel. Summing the number of directional "shadows" on a voxel identifies voxels as one that fills the ventricle chamber. Only voxels that have enough "shadows" should fill, so a filter removes voxels marked by only 4 or less shadows. A few gaps occur in the filled volume on the first try, so *Filler* is run twice to correct the gaps. A two-dimensional example of this procedure is shown on the following pages.

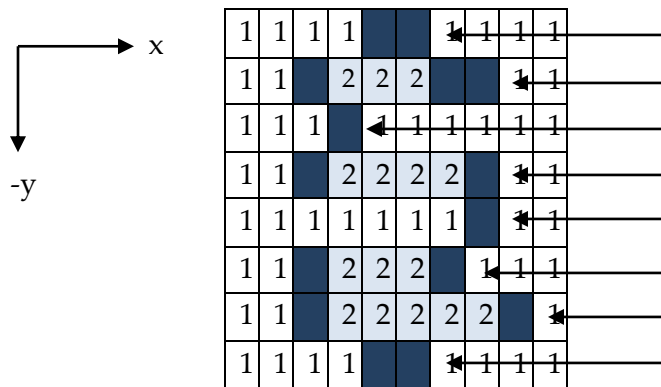
1. Ventricle Slice: Dark spaces are bright (>200) values. White spaces are zero value.



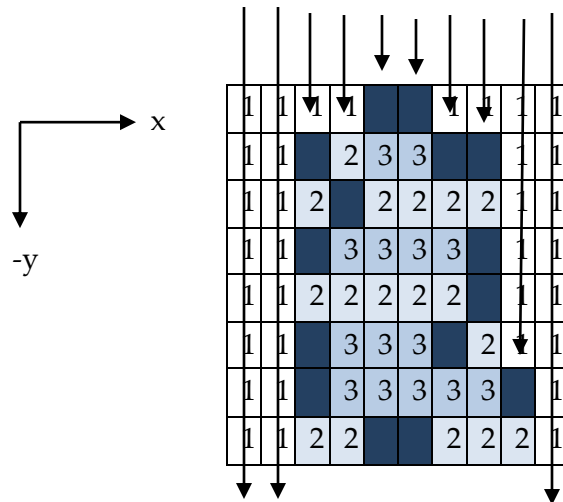
2. Shadow in the positive x direction. Add 1 to spaces after first bright value reached.



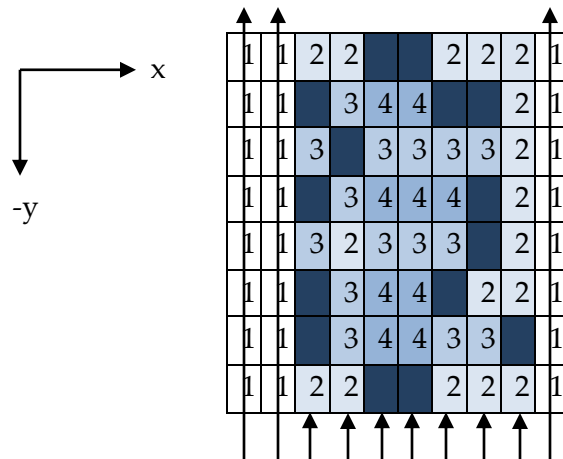
3. Shadow in the negative x direction. Add 1 to spaces after first bright value reached.



4. Shadow in the negative y direction. Add 1 to spaces after first bright value reached.

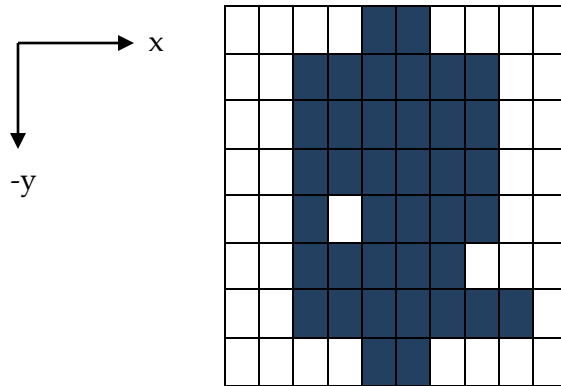


5. Shadow in the positive y direction. Add 1 to spaces after first bright value reached.

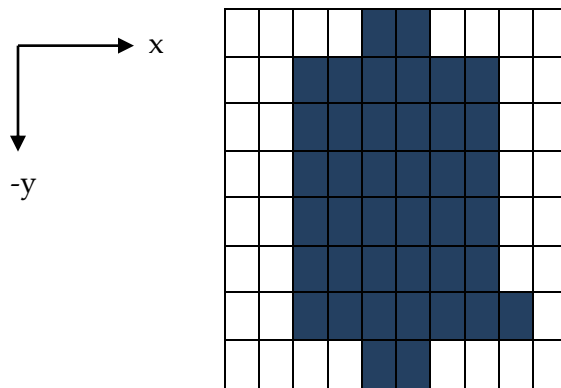


6. At this point, the interior pixels are a summation of the number of surfaces surrounding the pixels. A filter keeps only values of 3 or higher. Those remaining pixels are set to brightness of greater than 200.

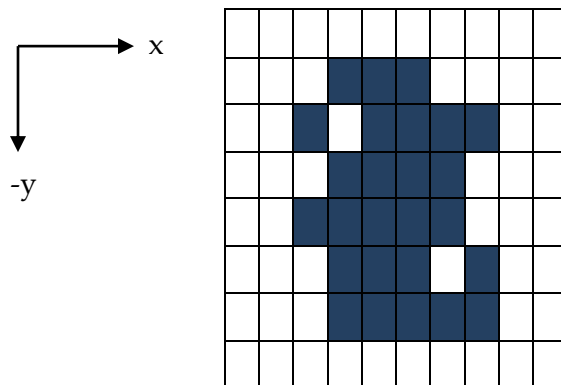
Result from previous page:



7. A few obvious pixels were missed. Filler steps 2 to 6 are repeated on this new image to fill those gaps.



8. Finally the original surface is removed. This avoids counting the ventricle wall.



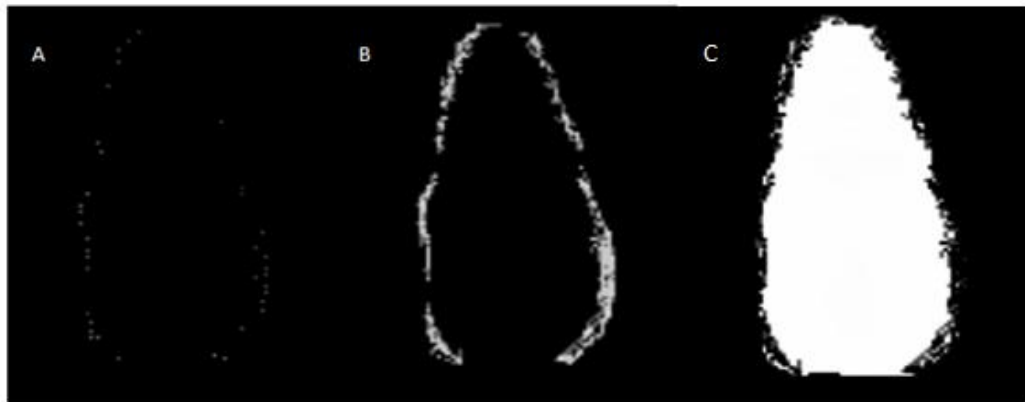
9. This final result is our filled image with a porous surface. There are 26 marked pixels.

Because there are 6 directions for shadowing three-dimensional volumes (+2 directions from 2D), the threshold for *Filler* is set to 4 (+1 shadow from 2D). If the threshold is set lower, pockets of incorrect voxels surrounding the surface add to the volume calculations. Examine Step 2 and imagine if the threshold were set to 2 instead of 3. Calculated volumes would overestimate real volumes. For the same reason, the original surface was removed. Ventricle volumes do not include the volume of the thin surface.

*Filler* achieves reasonable volume calculations if the *AddLine* surface is not littered with large gaps, where *AddLine* was unable to bind enough features together. *Filler* can be repeated several times to somewhat compensate. But on larger volume sets with more than twenty frames, *Filler* can take about ten seconds of runtime per iteration and the returns in volume rapidly drop off after a second iteration. Running *Filler* twice on the Selected Features volume is optimal.

## 4.9 Filler Algorithm Summary

In Figure 12, the results of each algorithm step are displayed on a 2D slice of the volume. In (A), the features are identified around a ventricular wall using the surface reconstruction model. In (B), *AddLine* has processed a surface from the tracked features. There are five noticeable gaps in the surface as well as the massive gap at the base of the ventricle. This gap is fixed by automatically introducing base features. Finally, in (C) the *Filler* function is run twice. *Filler* voxels are bright and occupy much of the ventricle chamber. The removed surface leaves pockets of marked voxels along the ventricle wall.



**Figure 18: (C) Filler has been applied to the surface in (B).**

Calibrating on balloons has made the algorithm viable for volume calculations on hearts. In the next section, the algorithm is used on the Normal Heart data set with four initial conditions on feature searching: (1) with the surface reconstruction model; (2) with the surface reconstruction model and base tracking; (3) with the automatic technique; (4) with the automatic technique and density filter.

## 5. Testing and Results

To test its general applicability, the algorithm was run on 7 total data sets: 3 balloons and 4 hearts. For these tests, the algorithm used the surface reconstruction model with base tracking. The algorithm was successful in producing reasonable results for all data sets.

For comparing the one surface model algorithm to the automated algorithm, participants inexperienced with 4DViz were invited to try the software. Five participants with varying knowledge of cardiac ultrasound were asked to trace a surface reconstruction model on the end-systolic frame of the Normal Heart image. They were given practice tracing with balloon data immediately beforehand. The average amount of time for tracing the 3D model on the ventricle was 5 minutes and 40 seconds. This is already a vast improvement over the hour it would have taken to trace an entire volume curve. For each of their models, the algorithm was run with 4 different search settings:

1. Using only the surface reconstruction model to search for features.
2. Using the surface reconstruction model and the automatic base feature search.
3. Using the automated spherical search with derivative identification of features.
4. Using the automated spherical search with derivative identification of features and a density filter to reduce extra-ventricular features.

## 5.1 Volume Results for Each Data Set

A surface reconstruction model is traced on the end-systolic frame of every balloon volume and every heart volume. Base features are added and then the algorithm is run to calculate volumes. The results are shown in Table 6. Ejection fraction is calculated by identifying the volume data's end-systolic volume and end-diastolic volume. These will respectively be the smallest and largest volumes calculated by the algorithm for each data set.

**Table 5: Volume and Ejection Fraction Results from Surface Model Technique**

DICOM file	Known Volume	Surface Model Volume	Algorithm's Calculated Volume	Ejection fraction result
55.6mL Balloon	55.6mL	55.6mL	57.6mL	5.8
83.8mL Balloon	83.8mL	92.0mL	77.9mL	4.3
110.0mL Balloon	110.0mL	131.32mL 121.32mL	122.7mL 114.4mL	4.9 5.5
Normal Heart	*52.86mL	58.47mL	56.31mL	47.3
Dilated Heart				32.2
Tilted Heart				30.5
Myopathy Heart				21.0

\* Value from hand drawn average of seven students

The algorithm volumes calculated on the balloons are promising. The 12% error on the 110mL balloon may be because the surface model was too large and features were found on the outside edge of the balloon. A second try with the 110mL balloon yielded

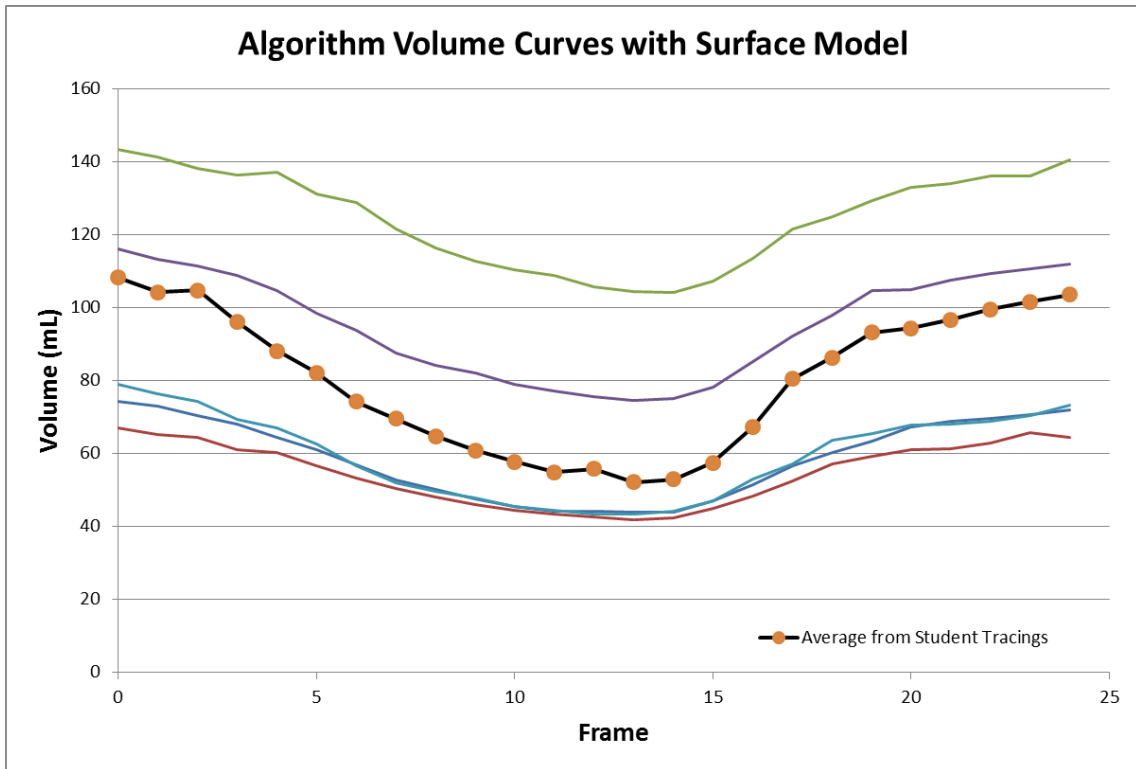
a more accurate result. This displays that algorithm volumes are dependent on the surface model drawing. The ejection fractions of the balloon are roughly 5% when they should be zero; the balloons did not change volumes frame to frame. The algorithm produced a 5% difference in the balloon's largest and smallest volumes.

The volumes of the heart data are unknown so there cannot be definite comparison to calculated volumes, but the ejection fractions are appropriate for the state of the hearts. None of these ejection fraction values are considered normal range, but the Normal Heart does best with a 47.3% ejection fraction compared to the 51.8% average ejection fraction from Figure 6. The 21% ejection fraction of the myopathy heart seems critically low, but when viewing the heart in 4DViz playback, the ventricle walls has hardly any motion. Finally, the algorithm did not seem to fail with the Tilted Heart, meaning results were not upright dependent. Overall, the surface reconstruction model technique produces promising results.

## 5.2 Comparison of Algorithm Techniques

How well did the automatic feature searching technique perform compared to the surface reconstruction model technique? This section discusses the results from the five participants and how each of four initial conditions altered the results.

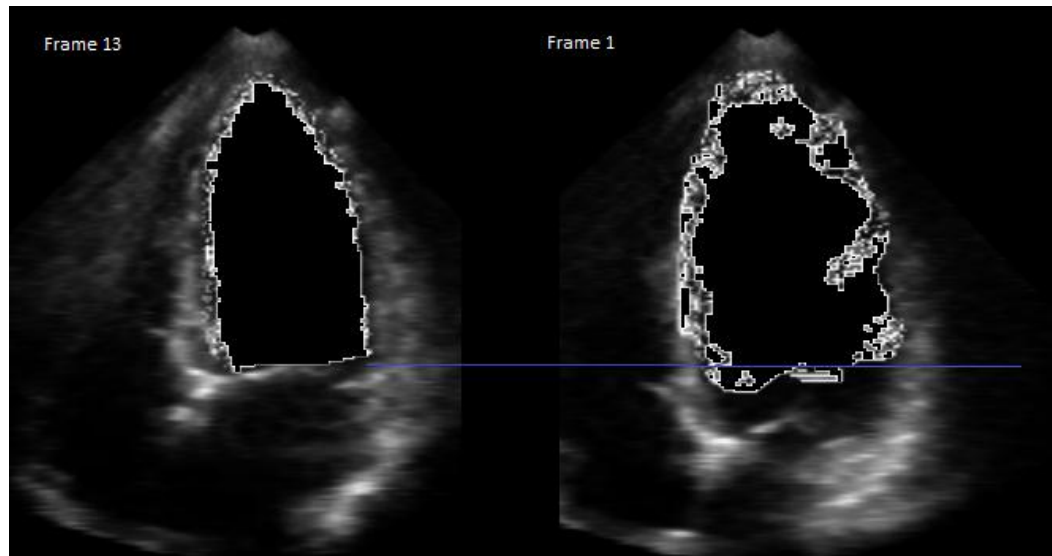
The first algorithm only applied feature searching using the surface reconstruction model. As expected, the user defined models led to greatly varying volume curves. The standard deviation of volumes at the same frame is 30.6mL, indicating disperse entries. The results are show in the figure below:



**Figure 19: Results of algorithm using only surface model. Volumes from 5 participants are widely varied around the hand drawn average and the slopes of the curves also do not match the hand drawn average.**

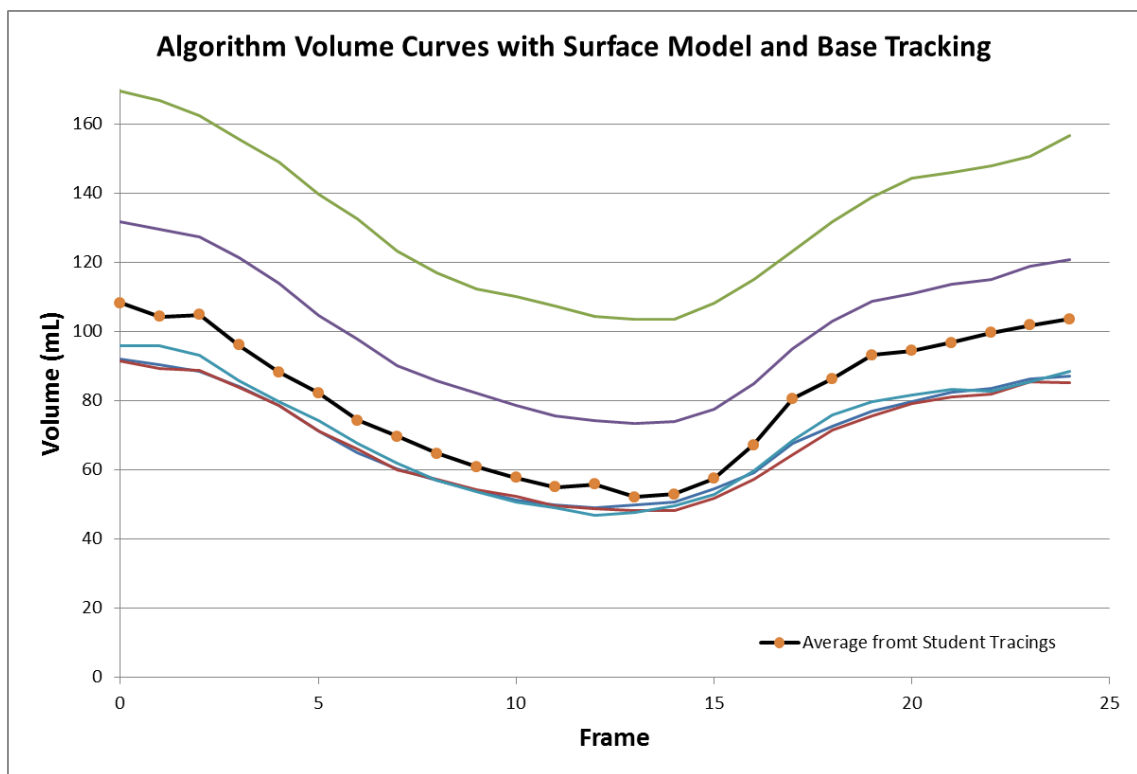
The average ejection fraction from this surface model algorithm is 37%, well below the 52% of the hand drawn average. Examining Figure 19 shows that the participants' volume curves did not match the student's drawn averages in shape. Although there is a decline in volume to systole and an increase in volume after systole, the slopes of the volume curves are not as steep. This accounts for the depression in ejection fraction.

Examining the 4DViz display of *Filler* ventricles created shows that the base is not adequately tracked from frame to frame. Fast relaxation of the ventricle walls is missed by the surface model's inability to track base features. The depressed slopes in the volume curves can be repaired if base features are added.



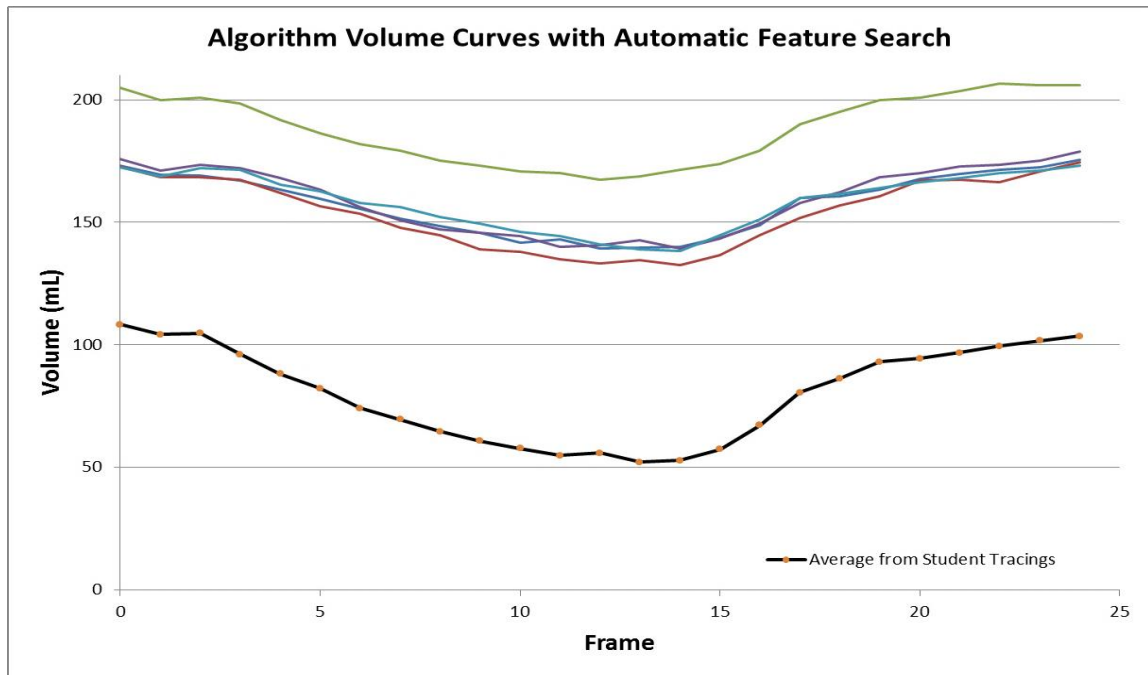
**Figure 20: Filled ventricle chamber matches on initial Frame 13, but by Frame 1, the filled space has not kept up with the base of the ventricle as seen by the blue line.**

By adding the base features to the surface model, the slopes on the volume curves become similar to the student's average. The volumes at the ends of the graph have been increased because the ventricle chamber is now filled to the base. The standard deviation does not improve significantly, but the average ejection fractions are improved to 46%.



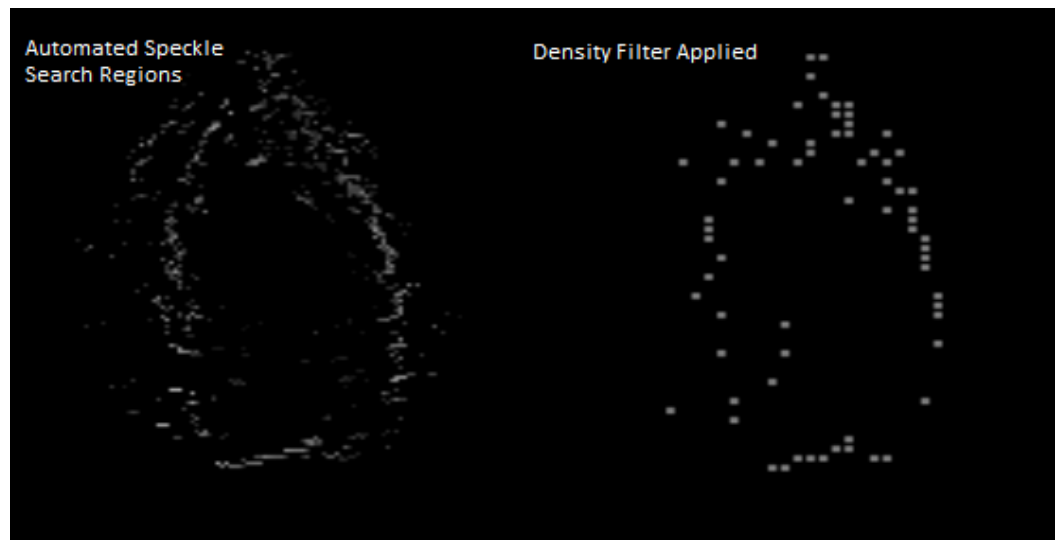
**Figure 21: Adding base features to the surface model improves the shape of the volume curve in matching the student's hand drawn average.**

Next, the automated features were used by the algorithm. With the unavoidable extra-ventricular features, the *AddLine* surfaces overestimate the ventricle wall and the *Filler* volumes overestimate the ventricular volumes. However, the volume curves are very consistent. The standard deviation is 13.9mL including one outlier.



**Figure 22: Volume curves generated by the automatic algorithm for feature searching. The participant volumes are fairly consistent, but are overestimated and do not have the shape of the student tracings.**

The ejection fractions are very low at 21% and the shape of the volume curves does not seem to match the average from the student tracings. Significant errors are apparent with the automatic method and are proportional to the amount of extra-ventricular features. A density filter is applied to reduce the inaccurate features; this can be seen in Figure 23. The density assisted feature search is the last set of initial conditions for the algorithm. Without any clear improvement from density filtering, the results of the automatic algorithm would be too inaccurate for application.



**Figure 23: The results of the density filter on the automated feature search. The density filter has removed much of the features outside the ventricle walls.**

The results of the density filter do improve the automated algorithm and lead to a possible next step for the algorithm. As expected, with fewer features outside the ventricle walls, the volumes have all decreased by about 20mL. In Figure 24, the participants' volume curves are more consistent with standard deviation of 6mL. The slopes of the volume curves are more similar to the student's average, and the ejection fractions have improved somewhat to 32%. This is still far from the 52% of the student average, but there is an interesting possibility brought up by the automated density volume curves. Because the volume curves are in a consistent range and have a shape similar to the goal, an offset can create an accurate volume curve. If 50mL is subtracted from every volume, the average ejection fraction improves to 53%. The offset volume curves are shown in Figure 25.

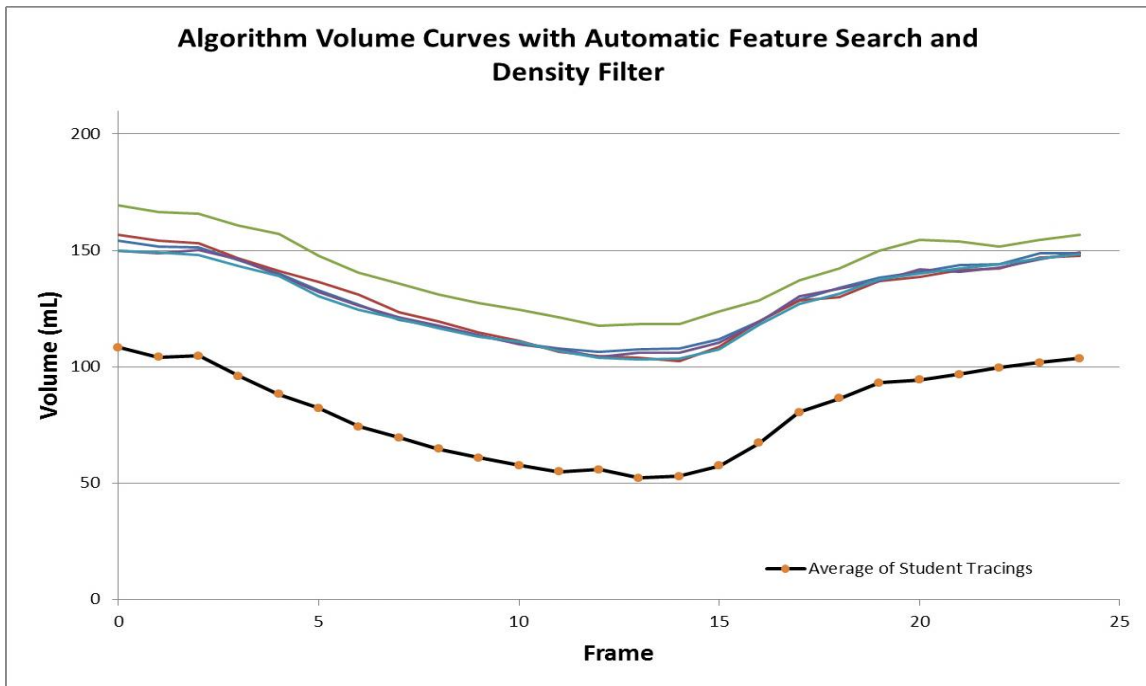


Figure 24: The density filter adjusts the shape of the volume curves to more closely match the average of the student tracings. The consistency of the participants' volumes is improved as well.

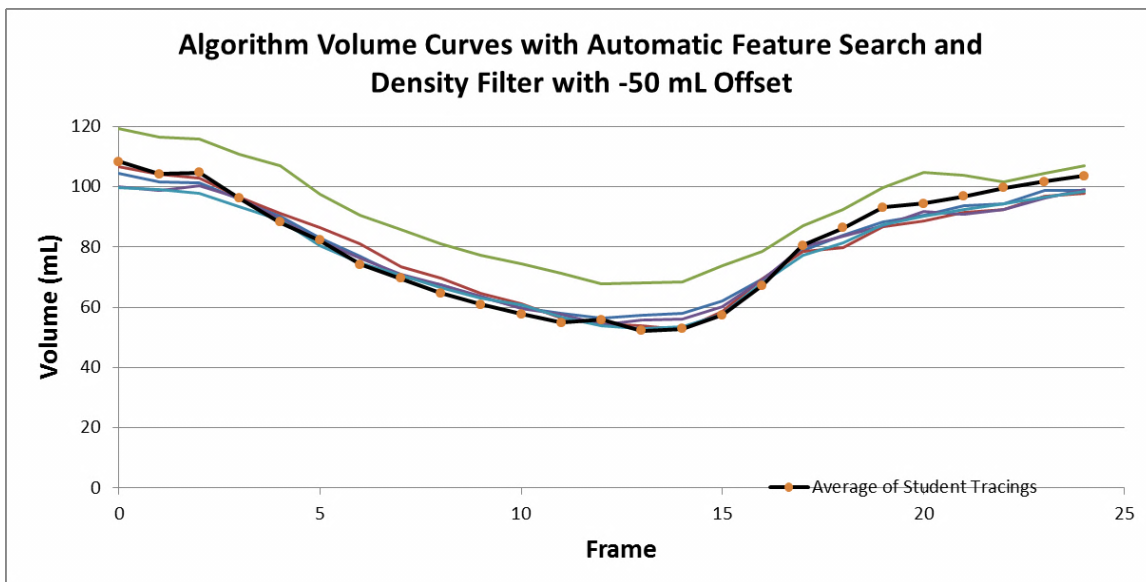


Figure 25: Volume curves from Figure 22 have been offset by -50mL. The resulting volume curves appear to accurately measure the average tracings.

At this time, the offset of 50mL is artificial. There is no supporting evidence that 50mL will work for other volume sets. The automated algorithm's results need to be repeated on other cardiac volume data sets to verify results. Another consideration is if there are ways for the algorithm to compensate for the 50mL offset, perhaps by improving the automated feature search or refining the density filter.

The surface reconstruction model with base tracking is capable of ejection fractions within 10% of the target. It is the best technique derived in this thesis, at the moment. The results show that the algorithm's automatic method is a promising and plausible way of calculating volume and ejection fraction in quick and consistent manner.

## 6. Conclusion

4DViz is an excellent experimental tool in testing the algorithm. Experimenting with a single parameter like SAD maximum search range, *AddLine* search radius, SAD threshold, or *Filler* iteration time can drastically alter resulting volume calculations. So it was useful just to be able to test the results of varying parameters. The surface reconstruction model was also used often during testing and for comparing results. 4DViz is robust enough software to test the image processing of ultrasound images.

Both single frame model and automatic *Filler* algorithms show promise in the area of ejection fraction calculation. Ohazama's surface model produced the most accurate volume results when improved with base tracking, but future work on the automated version can reduce user inputs and improve volume consistency. If ejection fraction measurements can be accurate to within 5%, then the algorithm may be of interest of cardiologists. For the algorithm to reach that point, experimenters must have known volumes for cardiac ultrasound data.

And, there are certainly many possible improvements at all levels of the algorithm. *AddLine* can be adjusted frame by frame. Base tracking features can be applied in their own histogram for a better understanding of their tracking efficiency. Run times were also never a large consideration in the programming, so time can definitely be improved upon. Once 4DViz is available, improvements can come from any source.

## Bibliography

- American Heart Association. (2011). *Heart Disease and Stroke Statistics*. Dallas, TX: Circulation Journal of American Heart Association.
- Bashford, G. R. (1995). *Three-Dimensional Ultrasound Speckle Characterization for Motion Detection*. Durham: Duke University.
- Christensen, D. A. (1988). *Ultrasonic Bioinstrumentation*. Salt Lake City: John Wiley and Sons.
- Craig, D. M. (1993). *Estimation of Right Ventricular Chamber Volume in Swine Based on a Geometric Model*. Durham: Duke University.
- Folland, E. D. (1979). Assessment of Left Ventricular Ejection Fraction and Volumes by Real-time, Two-dimensional Echocardiography. *Circulation Vol 60 No 4*, 760-766.
- Fraser, A. (2001). *Inge Edler and the Origins of Clinical Echocardiography*. Retrieved July 18, 2012, from <http://ehjcmaging.oxfordjournals.org>:  
<http://ehjcmaging.oxfordjournals.org/content/2/1/3.full.pdf>
- Geiser, B. P. (1999). *Patent No. 6106470*. United States of America.
- Gorcsan, J. (1993). *Comparison of left ventricular function by echocardiographic automated borderdetection and by radionuclide ejection fraction*. Pittsburgh: University of Pittsburgh Medical Center.
- Nelson, T. R. (1993). *Visualization of 3D Ultrasound Data*. San Deigo: University of California at San Diego.
- Ohazama, C. (1998). *Cylindrical Raster Map Based Surface Reconstruction of the Left Ventricle Using Real-Time 3D Ultrasound*. Durham.
- Philips. (2012). *iE33 xMatrix Echocardiography System Details*. Retrieved June 17, 2012, from <http://www.healthcare.philips.com>:  
[http://www.healthcare.philips.com/in\\_en/products/ultrasound/systems/ie33/](http://www.healthcare.philips.com/in_en/products/ultrasound/systems/ie33/)
- The Heart Foundation. (2012). *Heart Disease Scope and Impact*. Retrieved June 12, 2012, from The Heart Foundation: <http://www.theheartfoundation.org/heart-disease-facts/heart-disease-statistics/>