

LUND INSTITUTE OF TECHNOLOGY

MASTER'S THESIS

**Semi-Automatic Segmentation of the
Right Ventricle in MR Images**

Author:
Gabriella Gleisner

Supervisor:
Einar Heiberg

Examiner:
Anders Heyden

*A thesis submitted in fulfillment of the requirements
for the degree of Master of Science*

in the

Department of Clinical Physiology
Cardiac MR Group

April 12, 2017

Abstract

Cardiac magnetic resonance is widely used to non-invasively image the heart, for both clinical and research purposes. Segmenting different parts of the heart provides important diagnostic information. Performing this task manually is time-consuming and operator dependent, hence it would be beneficial to automate this process.

The aim of this thesis is to implement an algorithm to segment the right ventricle in cardiac magnetic resonance images. The right ventricle is geometrically complex, which makes it complicated to introduce prior knowledge to aid the segmentation. Therefore, the chosen segmentation approach was the Active Shape Model (ASM). Based on training data, ASM creates a shape model that will adapt to features in an image, making it likely to segment the object of interest.

The project was conducted at the company Medviso AB, and all implementations were done in the software, Segment, a platform for cardiovascular image analysis. The data used included image sets from 115 subjects; 53 heart patients and 62 healthy volunteers. 20% of these image sets were randomly selected as test data, while the rest were used to train the shape model. As validation, segmented volumes were compared to the volumes from manual delineation, and the Dice-Sørensen coefficients were computed. The results were generally good for the mid-ventricular slices, with some errors in the most apical slices.

In conclusion, the presented approach is promising for right ventricle segmentation, but will need further improvements before final implementation in Segment and clinical usage.

Populärvetenskaplig sammanfattning

Utlinjering av höger kammare i magnetisk resonanstomografi-bilder

Att tydligt urskilja olika delar av kroppens vävnader kan bidra till ökade kunskaper om fysiologin men också vara behjälpligt vid medicinska undersökningar. För att bidra till effektivisering av denna process presenteras följande arbete där en metod för utlinjering av höger kammare har utformats.

Magnetisk resonanstomografi är en av många metoder för att undersöka kroppens inre utan kirurgiska ingrepp och att kunna visualisera ett hjärta möjliggör nya sätt att studera kroppens viktigaste muskel. Hjärtat pumpar runt blod i kroppen och ser på så sätt till att nytt syre hämtas upp hos lungorna innan det transporteras ut till resterande organ i kroppen. Denna process är livsviktig och dess effektivitet är direkt påverkad av hjärtats funktionalitet och potentiella hjärtbesvär. En tydlig avbildning av hjärtat kan till exempel hjälpa läkare att diagnostisera sjukdomar och forskare att få en bättre förståelse av hjärtats fysiologi.

Fokus har i detta projekt varit höger kammare, ett av de fyra hjärtrummen. Från denna kammare pumpas blodet till lungorna via lungartären. Höger kammare har en relativt komplex form, som påminner om en halvmåne, men med stor variation mellan individer.

För att bestämma storleken på höger kammare i en magnetisk resonanstomografibild behöver man utlinjera den, vilket innebär att man markerar dess kontur för att kunna beräkna dess volym. Om detta görs manuellt är det tidskrävande och kan ge ett resultat som varierar beroende på erfarenheten hos den som utfört utlinjeringen. För att förenkla denna process är det fördelaktigt att använda automatiska metoder. Active shape-modellen är en iterativ och anpassningsbar metod och dess egenskaper gör den lämplig att använda för denna typ av problem, varvid det fokuserades på denna metod i detta projekt.

Genom att studera bilder med kända utlinjeringar av höger kammare hittar modellen olika skillnader och likheter mellan de kända formerna. Metoden skapar en medelvärdesform och genom träning lär den sig hur den kan variera denna form. För att sedan utlinjera höger kammare i en ny bild undersöker modellen egenskaper i den okända bilden och deformerar medelvärdesformen iterativt tills dess att höger kammare är funnen.

För att undersöka styrkor och svagheter hos den utformade modellen validerades resultatet genom att kända manuella utlinjeringar jämfördes med de utlinjeringar som den automatiska modellen erhöll. Resultaten var generellt bra, med en lite större felmarginal för bilderna från hjärtats nedre snitt. Slutsatsen är att den utformade modellen har utrymme för förbättringar men är en lovande grund för vidareutveckling av metoden.

Contents

Abstract	i
Populärvetenskaplig sammanfattning	ii
1 Introduction	1
2 Background	3
2.1 Human heart	3
2.2 Medical imaging	4
2.2.1 Cardiac imaging	4
2.2.2 Magnetic resonance scanning	4
2.3 Software	6
3 Aim of the Thesis	7
4 Theory	8
4.1 Mathematical background	8
4.1.1 Curve orientation	8
4.1.2 Procrustes Analysis	9
4.2 Image segmentation	10
4.2.1 Fast marching method	10
4.2.2 Active shape model	11
4.3 Validation	13
4.3.1 Dice-Sørensen coefficient	13
5 Implementation	15
5.1 Data set	15
5.2 Preparation of data	16
5.2.1 Orientation of the points	17
5.3 Training phase	17
5.3.1 Aligning the shapes	17
5.4 Segmentation	19
5.4.1 Initialization	19
5.4.2 Create edge map	21
5.4.3 Principal component analysis	23
5.4.4 Applying the model on mid-ventricular slices	23
5.4.5 Applying the model on remaining slices	24
5.5 Attempted approach	24
5.5.1 Principal component analysis	24
5.5.2 Applying the model	25
5.6 Validation	26
6 Results	27
6.1 Example segmentations	27

6.2	Validation results	29
7	Discussion and conclusion	34
7.1	Discussion	34
7.2	Attempted 3D approach	37
7.3	Limitations and future work	38
7.4	Conclusion	39
	Bibliography	40

Chapter 1

Introduction

Magnetic resonance imaging, MRI, has been used for medical purposes since 1977, when the first images of a human were produced by Raymond Damadian [1]. The first image showed a rough depiction of a thorax, and started an evolution of a new medical imaging technique. During the following years, different fields of applications were tested and the importance of MRI was realized. The development for a commercial product started, and the first whole-body scanner entered the market in 1980. From here on, the progress continued and the first images of a heart was achieved in 1981 [6]. These images contained a lot of noise and blurring, much due to cardiac and respiratory motion. Today, numerous techniques have been developed to overcome this problem. One method is to use electrocardiography gating to gather electrical activity of the cardiac cycle. Doing this over several heart beats help achieve images from each phase of the cardiac cycle [17].

Cardiac MR images have a wide range of use within medical research and clinical use. They can help give a better understanding of the heart muscle and its functions in a non-invasive and safe way for the patient. To easily compare certain areas, or make further computations, delineation of different parts can be useful. This process is called *segmentation* and is time-consuming if done manually. The resulting segmentations can also vary, depending on the judgment and experience of the person performing it. Automatic methods can thus help to create reliable and comparable results and also save time and money.

Today, there are several methods to automatically segment the left ventricle [21]. Multiple methods have been implemented and tested, using both image-based techniques and deformable models, giving various results. A very promising approach is to use a combination of mentioned methods (for further reading, see [8]). The ellipsoidal shape of the left ventricle makes it suitable for automatic segmentation. The right ventricle has a crescent shape, when viewed in the direction of the hearts shorter axis, and a triangular shape when viewed in the longer axis of the heart. The complex structure of the right ventricle makes it a harder task to automatically segment. The left ventricle has also previously been considered more *important* than the right ventricle, since it is the subject of pumping blood to the whole body. The right ventricle is merely pumping blood to the lungs and thus has not been in the center for the same amount of research. Hence, the number of articles about right ventricle segmentation is much less abundant than

the amount written about left ventricle segmentation. Nevertheless, efforts have been made with more or less successful results [18].

Delineation of the right ventricle in MR images is today manually performed in clinical routines and reliable automatic methods are requested. Thus, Medviso AB offered a master thesis opportunity with the ambition of creating a segmentation algorithm for this specific purpose. The project has been conducted in collaboration with the Lund Cardiac MR Group at Lund University, a research team with both engineers and physicians. The future goal is to implement an automatic algorithm in Segment, the Medviso created software for cardiovascular image analysis.

Chapter 2

Background

2.1 Human heart

As a vital muscular organ inside the human body, the heart pumps blood through the vessels, the body's circulatory system, to provide the body with nutrients and oxygen as well as removing metabolic waste from the cells [10]. After travelling through the body, the oxygen-depleted blood returns to the heart through the veins and ends up in the right atrium before continuing to the right ventricle. Here, the blood is pumped to the lungs, through the pulmonary artery, to retrieve fresh oxygen. The left atrium then receives the oxygenated blood, which then enters the left ventricle before being pumped out to the rest of the body through the aorta. The blood flow through the heart and the heart's anatomy is illustrated in Figure 2.1.

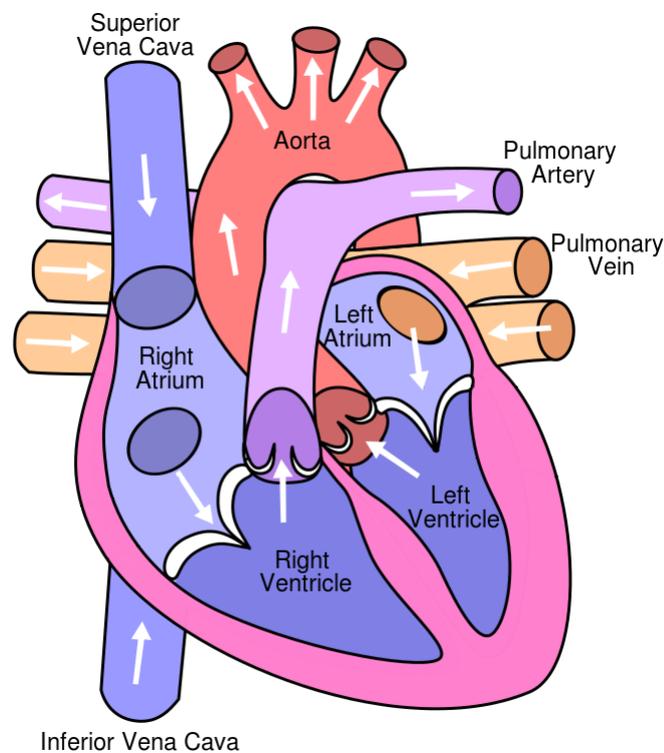


FIGURE 2.1: Schematic illustration of the human heart and its parts. Image adapted from [24].

To pump the blood through the body, the heart muscle alternately relaxes and contracts, allowing its chambers to fill up with blood and then force it out. The state of relaxation is when the heart has its maximum volume and is referred to as *end-diastole*, while the state of contraction is when the heart has its minimum volume and is referred to as *end-systole*. The walls of the cardiac muscle are called *myocardium*, where the outer and inner walls are referred to as *epicardium* and *endocardium* respectively. Being subject to much less pressure, and only pumping blood to the lungs, the myocardium of the right ventricle is much thinner than the walls of the left ventricle, which have to withstand a much higher load while pumping blood out to the whole body [10].

2.2 Medical imaging

Making it possible to visually represent the inside of the human body is an important noninvasive technique to help doctors and scientists. Medical imaging can, for example, help diagnose and choose treatments for diseases. It also enables potential databases of normal anatomy to help researchers get better understanding of physiology and to easier identify potential abnormalities, etc.

2.2.1 Cardiac imaging

There are multiple different imaging methods for medical purposes, and they all have their advantages and disadvantages. For cardiac imaging, a method that clearly depicts soft tissues is preferable. Some medical image modalities are also able to image in planes that are angulated to the hearts long and short axis. Short axis images show a cross section of the left and right ventricles, in a series of slices perpendicular to the long axis as shown in Figure 2.2.

The topmost part of the heart is referred to as the *basal* part and the bottom part is referred to as the *apical* part. Cardiac images on the short axis are usually sorted from the basal to the apical slices [10].

2.2.2 Magnetic resonance scanning

A magnetic resonance, MR, scanner (Figure 2.3) applies a magnetic field on a body placed in the scanner. Different magnetic characteristics of the tissues within the body makes it possible to depict the various structures [19].

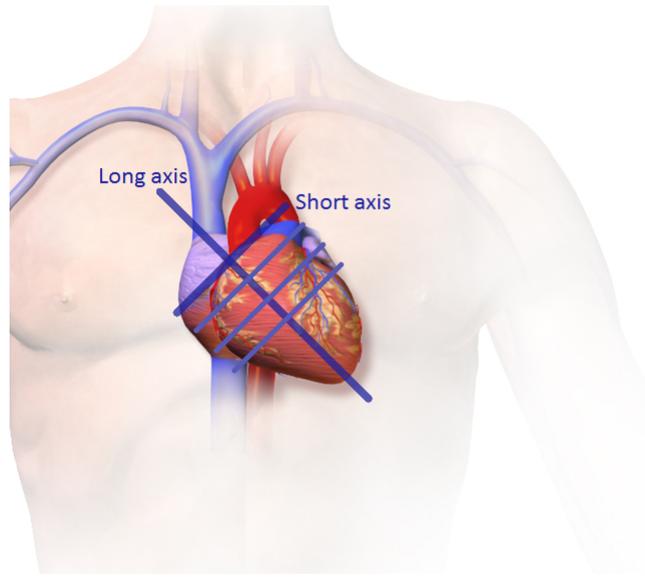


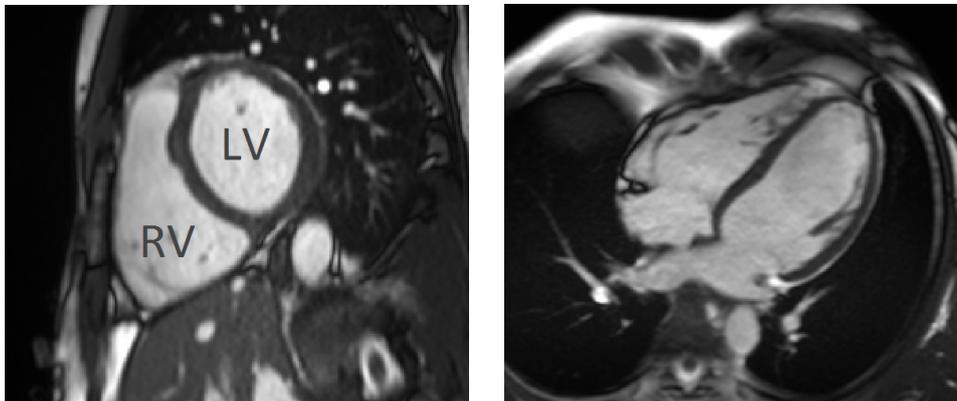
FIGURE 2.2: Illustration of the long and short axis of the heart. Image adapted from [2].

The scanner contains a large superconductor magnet, which causes the atoms in the body to align with the magnetic field. Low-frequency electromagnetic energy is then applied, alternately turned on and off, causing the atoms to change their directions before quickly returning to their relaxed state. Returning to the equilibrium, the atoms transmit their excess energy as radio waves, which are caught by a receiver. A computer is then able to analyse the data and create an image of the information from the radio signals. Because of the magnetic properties of particularly hydrogen atoms, which exist in abundance in fat and water, MR imaging distinguishes soft tissues very well.



FIGURE 2.3: A magnetic resonance scanner for medical use [9].

The electromagnetic field applied on the patient is created by gradient coils. MR images show cross sections of the internal body without harming the patient and by using gradient coils in multiple directions it is possible to image the patient in desired directions, such as long and short axis planes depicted in Figure 2.2 and 2.4.



(A) Short axis view. The right ventricle can be seen to the left and the left ventricle can be seen to the right.

(B) Long axis view, showing a cross section of all four chambers.

FIGURE 2.4: Example of cardiac imaging, captured with an MR scanner.

2.3 Software

Segment is a program developed in MATLAB by the Lund Cardiac MR Group at Lund University and is a validated software for cardiovascular image analysis [7]. Segment is freely available in a source code format for research purposes. For clinical use, the spin-off company Medviso sells two commercial versions, Segment CMR and Segment CT. The first version developed was Segment CMR for cardiac MR imaging, and later came Segment CT for computed tomography imaging. Both versions share the same platform, thus making all algorithms available from both versions.

Chapter 3

Aim of the Thesis

The aim of this thesis is to develop a semi-automatic segmentation algorithm for the right ventricle in MR images of the heart, based on the active shape model method. The goal is to use as few inputs as possible from the user. The focus of the segmentation is on one time-frame, the end-diastole, in short axis images. Due to the small thickness of the right ventricle myocardium, only the endocardium will be segmented. The results should be close to the manual segmentations, the computational time should be reasonable for clinical use and the final segmentations should generate satisfying result for the test data. The algorithm should also be implemented into Segment.

Chapter 4

Theory

Image segmentation is the task of partitioning an image into multiple segments, simplifying the analysing process of the image. The segmentation algorithm will be performed using the active shape model. To easier explain this process, this chapter will begin with a short mathematical background, explaining different techniques required in the active shape model, before explaining the main method. All images used in this thesis are in gray scale and are therefore represented as matrices with all values between 0 and 1, where 0 being the black pixels and 1 being the white pixels.

4.1 Mathematical background

4.1.1 Curve orientation

The orientation of an arbitrary curve can be found by analysing the sign of the angle at any vertex of the convex hull of the polygon. The convex hull of a curve is the minimal convex set still containing the curve, see Figure 4.1. Three sequential points on the edge of the convex hull make up two vectors in the plane, and the angle in between them is directly related to the sign of the cross product of the two vectors. Inserting the coordinates of the three points from the edge into an *orientation matrix*, \mathbf{O} , as seen in Equation (4.1), the sign of the cross product can be computed as the sign of the determinant of \mathbf{O} [23].

$$\mathbf{O} = \begin{pmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{pmatrix} \quad (4.1)$$

The determinant of \mathbf{O} will give a positive answer if the orientation is in the positive direction, and vice versa. If the answer is equal to 0 it means the three chosen points are in a straight line.

It is important to use the convex hull for this task because it removes the possibility of choosing points within a local concavity, since these might give a false answer.

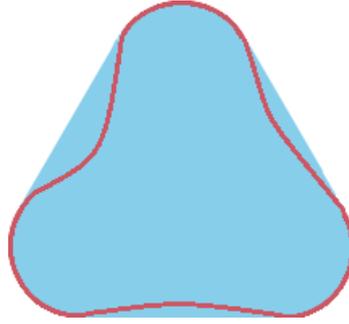


FIGURE 4.1: The blue area shows the convex hull of the red polygon, image adapted from [16].

4.1.2 Procrustes Analysis

Two shapes in the same plane can be aligned using rotation, translation and scaling through similarity transformation, a spatial mapping method where parallelism and angles are preserved.

In two dimensions, the orthogonal rotation matrix is given by

$$\mathbf{R} = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix}$$

where θ is the rotation angle. The translation vector is given by

$$\mathbf{t} = \begin{pmatrix} t_x \\ t_y \end{pmatrix}$$

where t_x and t_y is the movement in x and y direction respectively. Finally, since the scaling is given by a positive scaling scalar, s , the transformation is given through Equation (4.2) [3].

$$T(\mathbf{x}) = s\mathbf{R}\mathbf{x} + \mathbf{t} \quad (4.2)$$

To align one shape to another, the best matching transformation variables needs to be computed. This is achieved by minimizing the difference between the two shapes, i.e. finding \mathbf{R} , \mathbf{t} and s , that solves Equation (4.3), where \mathbf{u}_i and \mathbf{v}_i are the coordinates of the corresponding landmarks in two different shapes, and N is the number of landmarks in the shapes.

$$\min \sum_{i=1}^N \|\mathbf{u}_i - \mathbf{t} - s\mathbf{R}\mathbf{v}_i\|^2 \quad (4.3)$$

The unknown variables, \mathbf{R} , \mathbf{t} and s , in (4.3) can be estimated using so called *procrustes analysis*.

Starting by computing the mean values of the coordinates in \mathbf{u}_i and \mathbf{v}_i respectively, pursuant to

$$\bar{\mathbf{u}} = \frac{1}{N} \sum_{i=1}^N \mathbf{u}_i$$

$$\bar{\mathbf{v}} = \frac{1}{N} \sum_{i=1}^N \mathbf{v}_i$$

before introducing two new variables

$$\tilde{\mathbf{u}}_i = \mathbf{u}_i - \bar{\mathbf{u}}$$

$$\tilde{\mathbf{v}}_i = \mathbf{v}_i - \bar{\mathbf{v}}$$

Let

$$\mathbf{H} = \sum_{i=1}^N \tilde{\mathbf{v}}_i \tilde{\mathbf{u}}_i^T$$

and use singular value decomposition to get $\mathbf{H} = \mathbf{U}\mathbf{D}\mathbf{V}^T$.

Finally, the estimated variables, denoted $*$, are given by Equation (4.4), (4.5) and (4.6) (for further derivation see [15]). The transformation can then be done as shown in Equation (4.2).

$$\mathbf{R}^* = \mathbf{U} \text{diag}(1, 1, \det(\mathbf{U}\mathbf{V}^T)) \mathbf{V}^T \quad (4.4)$$

$$s^* = \frac{\sum_{i=1}^N \tilde{\mathbf{v}}_i^T \mathbf{R}^* \tilde{\mathbf{u}}_i}{\sum_{i=1}^N \|\tilde{\mathbf{u}}_i\|^2} \quad (4.5)$$

$$\mathbf{t}^* = \bar{\mathbf{v}} - s^* \mathbf{R}^* \bar{\mathbf{u}} \quad (4.6)$$

4.2 Image segmentation

The task of partitioning certain features in a digital image is called *image segmentation*. This is a procedure to enhance complexity and simplify the analysis of images. There are numerous methods for this process and the applications are, more or less, unlimited. For this thesis two of these methods were used, one numerical and one statistical, both of which will be described in the following section.

4.2.1 Fast marching method

The fast marching method can be used to numerically find an approximated solution to boundary value problems of the Eikonal equation [20], seen below.

$$|\nabla v(x)| = \frac{1}{f(x)} \text{ for } x \in \Omega$$

$$v(x) = 0 \text{ for } x \in \partial\Omega$$

Given an initial curve, v , the algorithm finds the arrival time from the curve to all other points within the concerning area using a constant speed, $f(x)$. This method is used for problems where the speed function is always propagating in the direction of the normal of the curve. This makes it possible to change the formulation to a stationary problem, making it a very fast approach to solve the problem. The method sweeps through the surrounding area of L points in $L \log L$ steps. The result is a mapping of the given curve, and its surrounding area, of the time it takes for the speed function to arrive at each point individually.

4.2.2 Active shape model

When trying to segment an object whose appearance can vary, it is beneficial to use an adaptable method. The *active shape model* builds a model, computed from training data, and tries to match the model to data in a new image. By comparing similarities and looking for variations in the training data, various possible deformations can be found and used to adjust the model to a desired object and the underlying image.

To create an active shape model, it initially needs to be trained using manually segmented images of the desired structure. These segmentations are preferably done by an expert in the field and contain various different shapes to cover a big spectrum of variations and appearances. This training data must also be consistent with the same number of points evenly distributed along the edge of the segmented structure and using the same starting position [22].

Assume the set of training data consists of M images, each with N corresponding landmarks, for which the coordinates are known. Initiating the algorithm, the training data must be aligned, for instance by using procrustes analysis.

For a model to be able to generate an arbitrary shape of the desired structure, common patterns and different variations among the shapes in the data set needs to be found. This could be done using *principal component analysis*, PCA, where a new basis that best represent the shapes will be found. The principal components are the underlying structure of the data and they will show the directions of the most variance in the landmarks. The N points in the set, with two coordinates (x and y) each, will then span the new space, hence the new vectors will represent the shapes in the $2N$ -dimensional space. These vectors will thus represent a new basis to use for creating the coveted shape model [4].

The first step of the PCA is to compute the mean shape of all the pre-segmented shapes, as done in Equation (4.7), where \mathbf{u} is the $N \times 2$ -matrix with the coordinates of the landmarks and j denotes shape number. The

deviation from the mean for each shape in the set is then computed as per Equation (4.8).

$$\bar{\mathbf{u}} = \frac{1}{M} \sum_{j=1}^M \mathbf{u}_j \quad (4.7)$$

$$\mathbf{d}\mathbf{u}_j = \mathbf{u}_j - \bar{\mathbf{u}} \quad (4.8)$$

The difference vectors, $\mathbf{d}\mathbf{u}$, can be used to find the covariance matrix, \mathbf{S} , as per Equation (4.9), where the $\frac{1}{M}$ -term makes sure \mathbf{S} is unbiased.

$$\mathbf{S} = \frac{1}{M} \sum_{j=1}^M \mathbf{d}\mathbf{u}_j (\mathbf{d}\mathbf{u}_j)^T \quad (4.9)$$

The final step of the PCA is to compute the eigenvalues and eigenvectors of \mathbf{S} , as in Equation (4.10).

$$\mathbf{S}\mathbf{P} = \lambda\mathbf{P} \quad (4.10)$$

The eigenvectors, \mathbf{P} , gives the dominant directions in the $2N$ -dimensional space, each of them representing a different variational mode in the training data. The eigenvalues, λ , will be presented in decreasing order and will each give the variance of data in the direction \mathbf{P} , ordered from the most to the least significant one. Since all eigenvectors are orthogonal, the total variance is given by

$$T = \sum_{i=1}^{2N} \lambda_i.$$

where i denotes the point number. Any of the original shapes can now be expressed as in Equation (4.11), where $\mathbf{P} = (\mathbf{p}_1 \mathbf{p}_2 \dots \mathbf{p}_{2N})$ is the transformation matrix and \mathbf{b} is a vector with $2N$ elements, denoting the coordinates of the shape in the new basis. \mathbf{b} can also be interpreted as the appropriate weights for each of the deformation vectors in \mathbf{P} .

$$\mathbf{u} = \bar{\mathbf{u}} + \mathbf{P}\mathbf{b} \quad (4.11)$$

It is required to use the principal components with most variation to cover the relevant deformations of the training data. Therefore, the K largest eigenvalues should be used to account for the majority of the total variance, since the last modes comes from the noise in the training data. Only using the K first vectors of \mathbf{P} thus gives an approximation of a shape, see Equation (4.12), which is the sought after shape model and completes the training process.

$$\mathbf{u} \approx \bar{\mathbf{u}} + \mathbf{P}_K \mathbf{b}_K \quad (4.12)$$

The variational modes are now found and the model can be applied to data in a new image. To be able to compute the still unknown \mathbf{b}_K , some information from the new image needs to be extracted. One method is to create a rough edge map, \mathbf{v} , of the structure aimed to be segmented, e.g. using the fast marching method. \mathbf{v} will present a good starting position for the model in the new image, and \mathbf{b}_K could be interpreted as the weights needed to fit the model in Equation (4.12) to the desired shape. Hence, \mathbf{b}_K is depending on the distance between the mean shape, $\bar{\mathbf{u}}$, and the new data, \mathbf{v} . Rewriting Equation (4.12), using the orthogonal properties of the vectors in \mathbf{P}_K and the rough approximated shape \mathbf{v} , the expression in Equation (4.13) is acquired [3].

$$\mathbf{b}_K = \mathbf{P}_K^T(\mathbf{v} - \bar{\mathbf{u}}) \quad (4.13)$$

To make sure the weights do not become too big, they are limited by the constraint in Equation (4.14) [4].

$$-3\sqrt{\lambda_K} \leq \mathbf{b}_K \leq 3\sqrt{\lambda_K} \quad (4.14)$$

After the deformations have been applied to the model, it is yet again aligned to the edge map before new displacements are computed and the algorithm is iterated until convergence.

4.3 Validation

4.3.1 Dice-Sørensen coefficient

The *Dice-Sørensen coefficient* is very suitable to use as a validation method for segmentations. It gives a statistic value of similarity for two samples, two segmentations in this case. The coefficient is given by Equation (4.15) where T_P denotes the true positive values in the segmentation, i.e. the correctly segmented pixels, F_P denotes the false positives, the part in the segmentation that is not in the original image, while F_N denotes the false negatives, the parts of the original image not within the segmentation [5].

$$c_{DS} = \frac{2T_P}{2T_P + F_P + F_N} \quad (4.15)$$

See Figure 4.2 as an example, where the red circle represents the true shape and the blue circle represents the segmentation. The true positives, T_P , are the pixels in the area within both circles coloured in purple. The light gray area displays the false positives, F_P , while the dark gray area displays the false negatives, F_N . The coefficient will be a number between 0 and 1, with 0 being no common area at all and 1 being complete coverage.

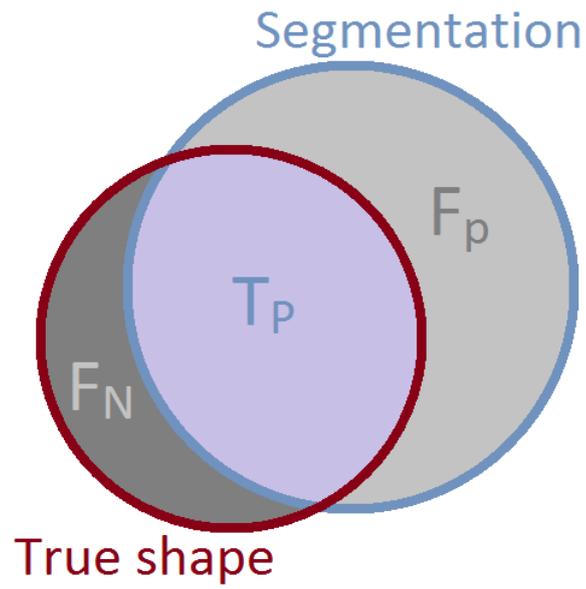


FIGURE 4.2: Schematic illustration of the different variables in Equation (4.15).

Chapter 5

Implementation

5.1 Data set

The data set used consisted of 115 short axis image sets, provided by the cardiac MR group at Lund University. An example of such a set can be seen in Figure 5.1, where the manual segmentations of the right ventricle can be seen as overlay in magenta. The data came from 62 healthy volunteers and from 53 consenting patients with various heart conditions. All images were manually segmented by a physician and the right ventricle could be found in between 9 and 14 slices of each set. From each of these groups were 20% of the data sets randomly selected to be used as test image sets, while the rest were kept as training data for the algorithm, resulting in 23 test sets and 92 training sets.

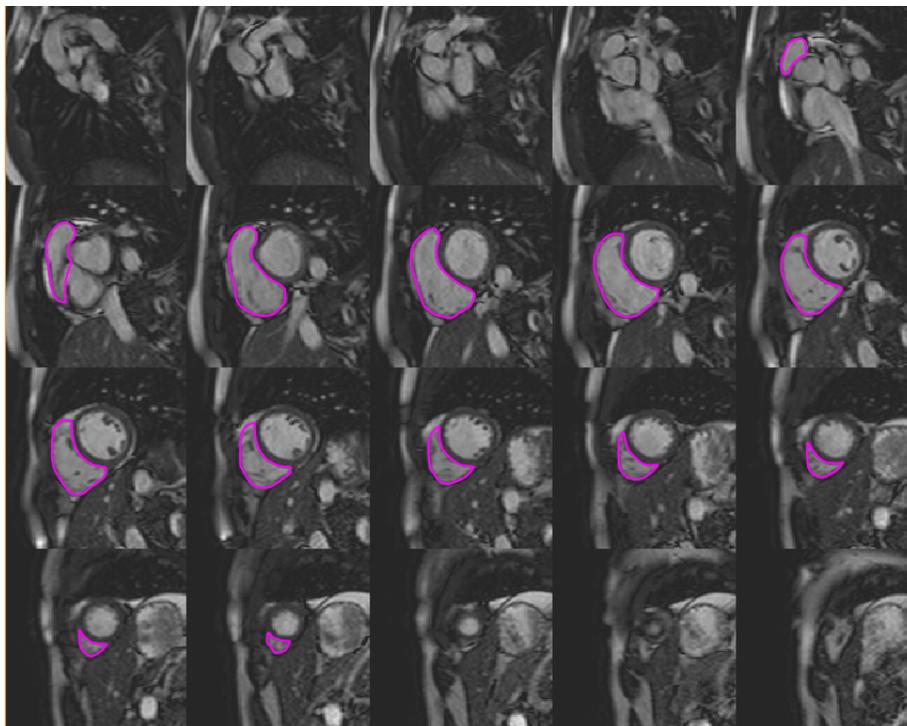


FIGURE 5.1: An example of a short axis image set, sorted from the base to the apex of the heart, showing manual segmentations of the right ventricle.

A challenge was to determine where the transition from the right ventricle to the pulmonary artery is taking place in the basal slices. In a few of the sets the most basal images show segmentations where there are no clear borders present and the manual segmentations have been completed using the physician's knowledge and previous experiences. Thus, it was decided to exclude the most basal slices due to this challenge.

5.2 Preparation of data

A manual segmentation of a mid-ventricular slice can be seen in Figure 5.2, where the left ventricle myocardium is marked with green (epicardium) and red (endocardium), while the right ventricle endocardium is segmented with a magenta coloured line. Due to the small thickness of the right ventricle myocardium, it is hard to distinguish both the endo- and epicardium. Hence, only the endocardium is segmented and therefore solely the endocardium is sought after in this thesis.

Using Segment, the coordinates of the manual segmentations of the right ventricle in all training sets were extracted.

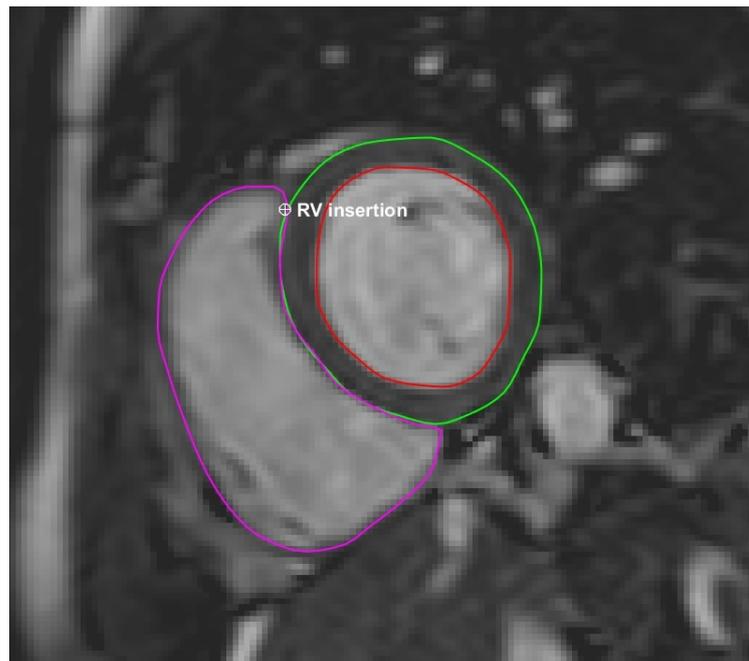


FIGURE 5.2: A mid-ventricular slice, where the epi- and endocardium of the left ventricle is segmented with green and red lines respectively, and the right ventricle endocardium is segmented with the magenta coloured line. The upper insertion point between the right and left ventricle can also be seen, marked with an annotation point.

The upper insertion point, between the right and left ventricle, were manually marked by the author in a mid-ventricular slice in all sets, using a built-in annotation point tool in Segment. The coordinates of these points were also extracted. The landmarks in each shape were sorted so the first one would be the one closest to the insertion point, and the rest following

in clockwise order, to ensure they all had the same starting position. Most segmentations consisted of 80 landmarks and those few shapes that did not were resampled to concur with the standard.

5.2.1 Orientation of the points

Since all shapes later will be compared point wise, it is important that they all have the same orientation of their points.

Using MATLABs function CONVHULLN [12], the convex hull of the current shape was found, see Figure 5.3a. Extracting the coordinates of three points known to be on the edge of the convex hull (Figure 5.3b), and inserting them into the orientation matrix, as seen in Equation (4.1), made it possible to compute the determinant and thus finding their orientation.

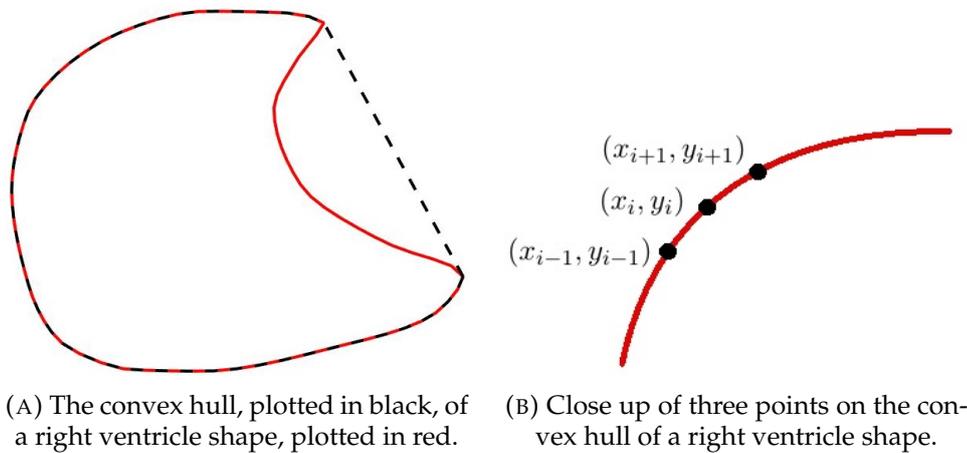


FIGURE 5.3

Since most of the segmented shapes in the training data were oriented clockwise, this was set as the default orientation. Those segmentations with a counter-clockwise orientation had their points rearranged to the correct order. If the determinant of the orientation matrix (Equation (4.1)) was equal to 0, the three chosen points were in a straight line and three new points on the same convex hull were chosen instead.

5.3 Training phase

The training phase is important for the active shape model, since it is used to gather and prepare all data making it possible to create a model. *A priori* knowledge about the structure and expertise from a physician are important contributions to the model, which both can be added during this phase.

5.3.1 Aligning the shapes

Since all image slices in a short axis set comes from the same heart, it is expected that they are related to each other and have the same orientation

within the body. The shapes in different slices can vary a lot from set to set, but the mid-ventricular slices are quite similar. Thus, the initial alignment was solely performed on the mid-ventricular slices, in which the right ventricle insertion point was applied earlier. This also contributed to reducing the computational time. The segmentations from these slices were extracted separately from the training data. An overview of the aligning process can be seen in Figure 5.4.

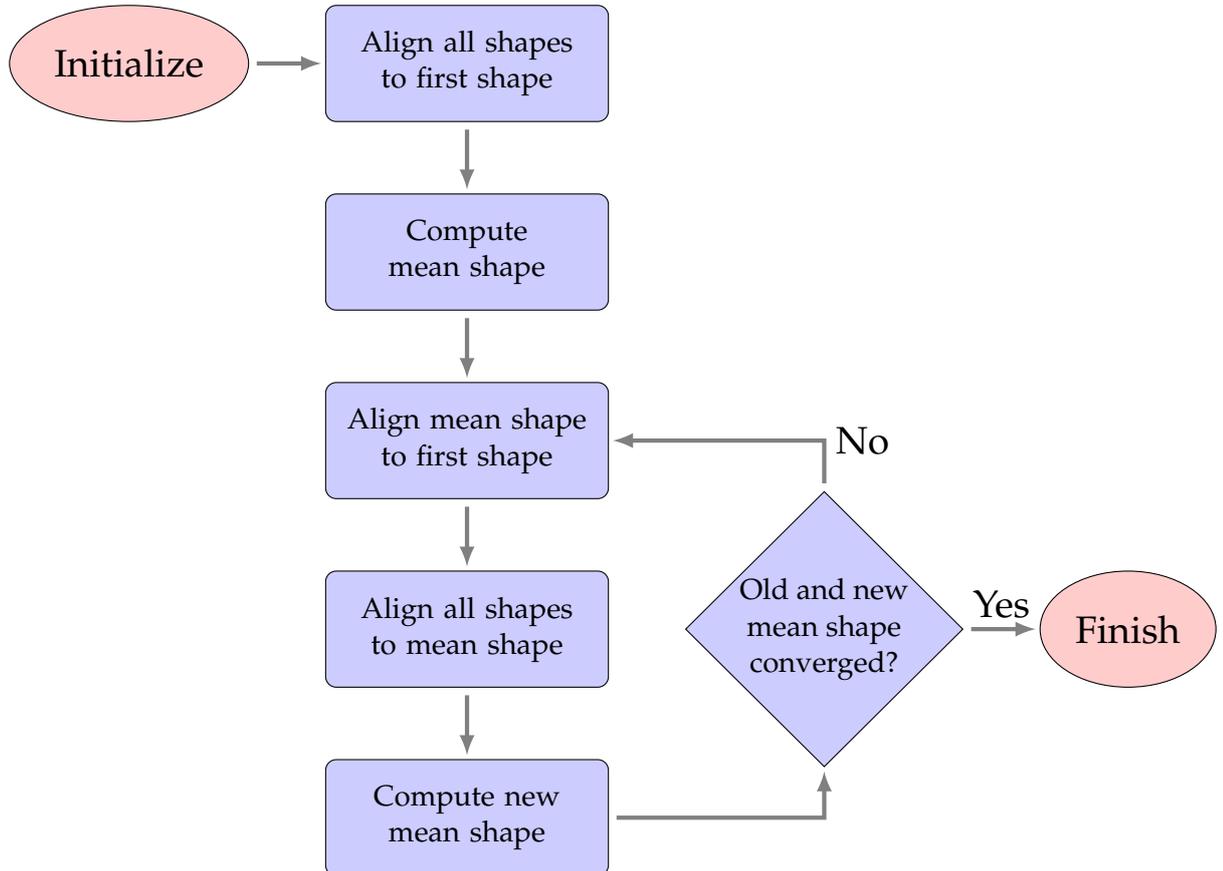
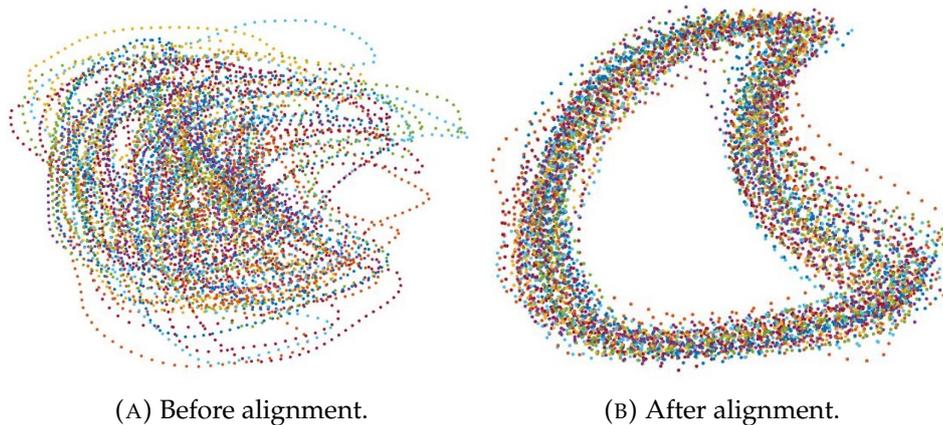


FIGURE 5.4: Flow chart showing the iterative process of aligning the mid-ventricular right ventricle shapes.

The mid-ventricular shapes were first aligned to the first shape in the set. All alignments were made by finding the variables \mathbf{R}^* , \mathbf{t}^* and s^* that best solves Equation (4.3), as described in Section 4.1.2. A mean shape was then computed by calculating the mean value for each point, as in Equation (4.7). To ensure convergence, the mean shape was then aligned to the first shape as well, before letting all shapes in the set align to the mean shape. A new mean shape was then computed and this process was iterated until convergence, i.e. when the difference between the old and new mean shape was practically 0. All mid-ventricular slices, before and after alignment, can be seen in Figure 5.5.



(A) Before alignment.

(B) After alignment.

FIGURE 5.5: All mid-ventricular shapes from the training data.

The resulting \mathbf{R}^* , \mathbf{t}^* and s^* after all iterations, were saved for each slice. These were later used to transform the rest of the slices from the same image set as the corresponding mid-ventricular slice.

5.4 Segmentation

The segmentation problem was approached treating each slice separately in two dimensions (2D), thus creating a model for each slice. An overview of the complete segmentation process can be seen in Figure 5.6.

5.4.1 Initialization

Some input was required from the user to be able to execute the segmentation on the test data. An image set from the test set was loaded in Segment and, as in the preparation step of the training data, the right ventricle insertion point was marked in one of the mid-ventricular slices. A seed point was also placed somewhere in the middle of the right ventricle, marked + in Figure 5.7.

By looking at a cross section image from a long axis view, the user also established the first and last slice in the short axis set that included the right ventricle. The total number of slices desired to be segmented in the test image set was then computed and all sets in the training data were interpolated to contain the same number of slices.

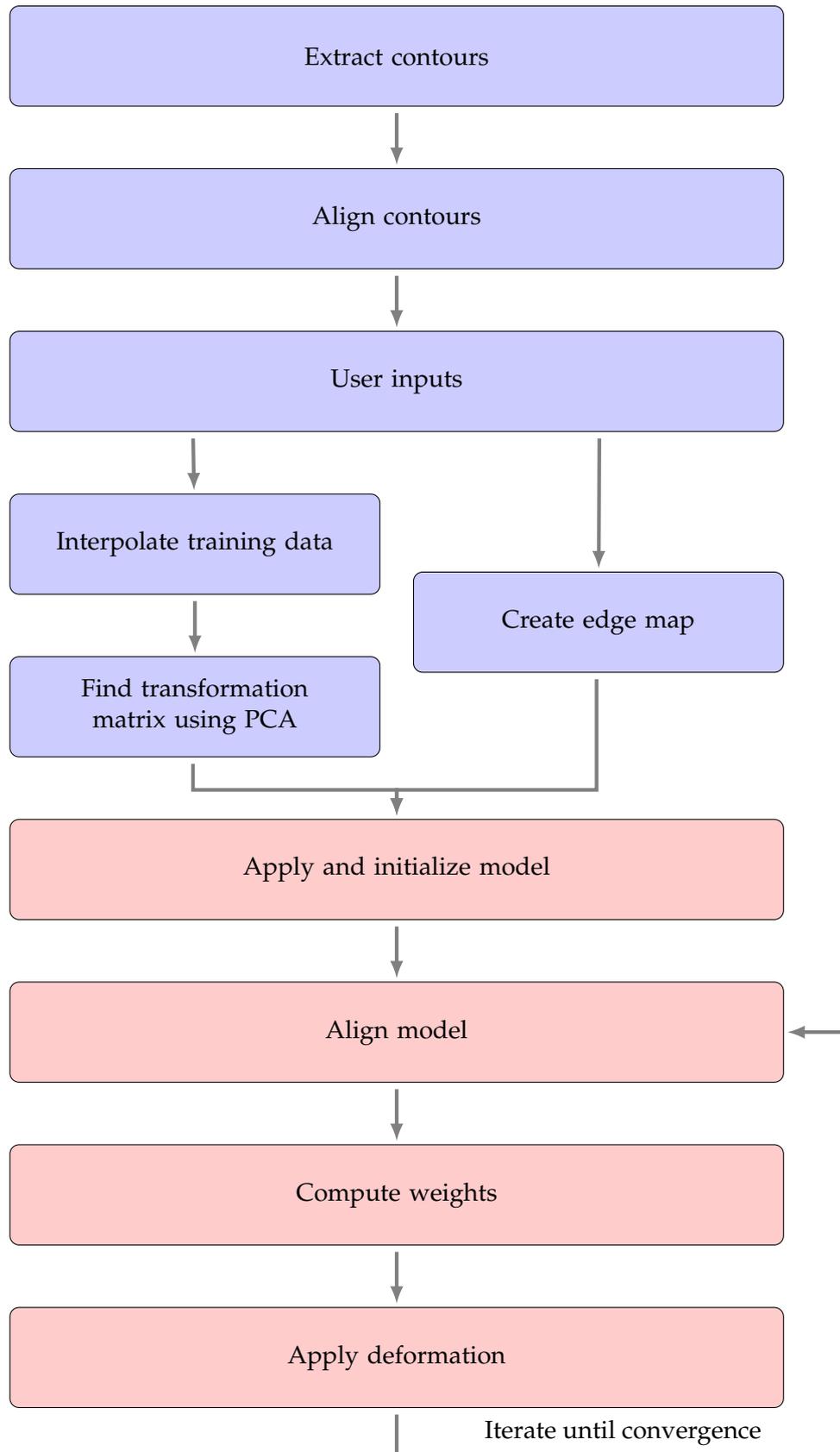


FIGURE 5.6: Flow chart showing the basic steps of the complete segmentation process, blue boxes denoting the training process and red denoting the segmentation process.

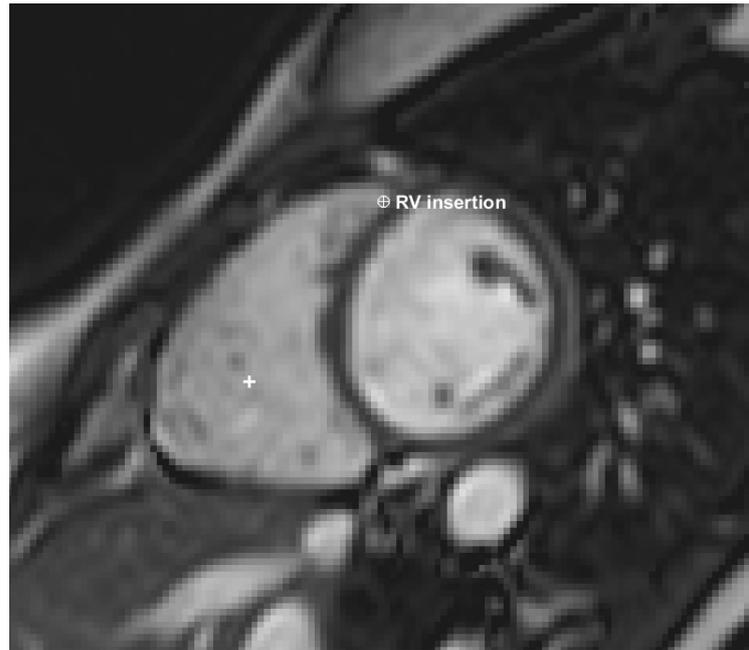
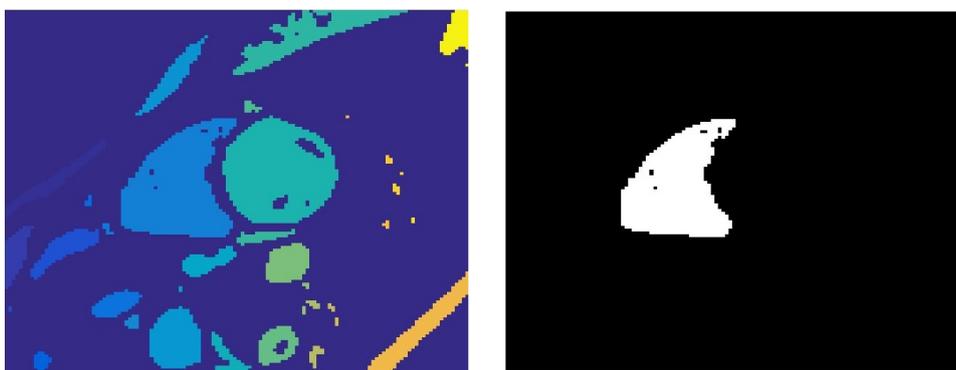


FIGURE 5.7: A mid-ventricular slice with the manually placed inputs marked; the right ventricle insertion point and the seed point.

5.4.2 Create edge map

The edge map for the mid-ventricular slice can be created after the initial inputs have been made. First, the current image is converted to a binary image, using the MATLAB function `GRAYTHRESH` [13], to find a normalized intensity value of the image to be used as a threshold. Applying `BWLABEL` [11], an image is given showing all different structures in the input image, all marked with different labels, as the example in Figure 5.8a. The structure containing the seed point is extracted, which gives a binary image, just containing the right ventricle, as seen in Figure 5.8b.



(A) An example of a labeled binary image.

(B) The right ventricle extracted from the binary image.

FIGURE 5.8

To get rid of possible, but unwanted, holes in the structural shape of the right ventricle, as well as creating smoother edges, fast marching is performed on the image, using the seed point as the starting point. Once the arrival time is obtained, see example in Figure 5.9, the intensity at the position of the right ventricle insertion point was used as a threshold, and the resulting shape was found.

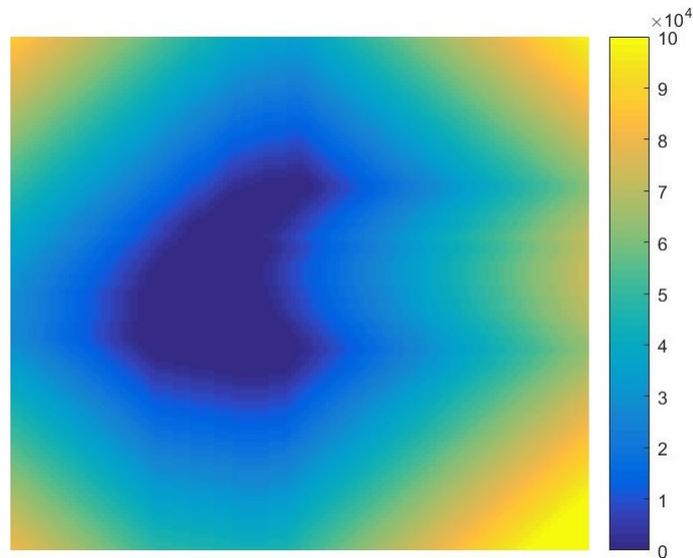


FIGURE 5.9: The arrival time displayed in a plot, showing a distinct right ventricle structure.

The edge was extracted from the resulting shape before the orientation was inspected to be in the correct direction and the output was interpolated to contain the default number of points. An example of a mask of an edge map can be seen in Figure 5.10.

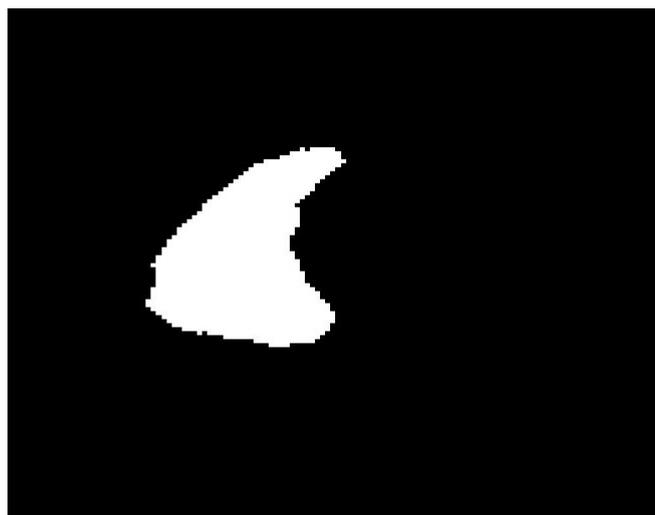


FIGURE 5.10: A mask of an edge map example from the test image set.

5.4.3 Principal component analysis

To find the deformation parameters for the model, principal component analysis was performed on all sets of slices, one at a time, as the last part of the training phase. First, the covariance matrix for each set was computed, as explained in 4.2.2. Using Equation (4.10), the transformation matrices, \mathbf{P}_j , were computed, as well as the eigenvalues, λ_j .

Since this process was performed in the $2N$ -dimensional space, where N denotes the number of points, which as a default is 80, \mathbf{P} will contain 160 deformation vectors and correspondingly the number of eigenvalues will be 160 as well, for each slice.

In order to choose the amount of deformation vectors to use in the transformation matrices, the eigenvalues were observed. Figure 5.11 shows an example of the 50 first eigenvalues from a mid-ventricular slice in one of the test images. It can easily be seen how most of the eigenvalues are approximately 0 after the rapid decrease in the beginning. To make sure enough variational modes were used, it was decided to use 20% of the vectors in \mathbf{P} .

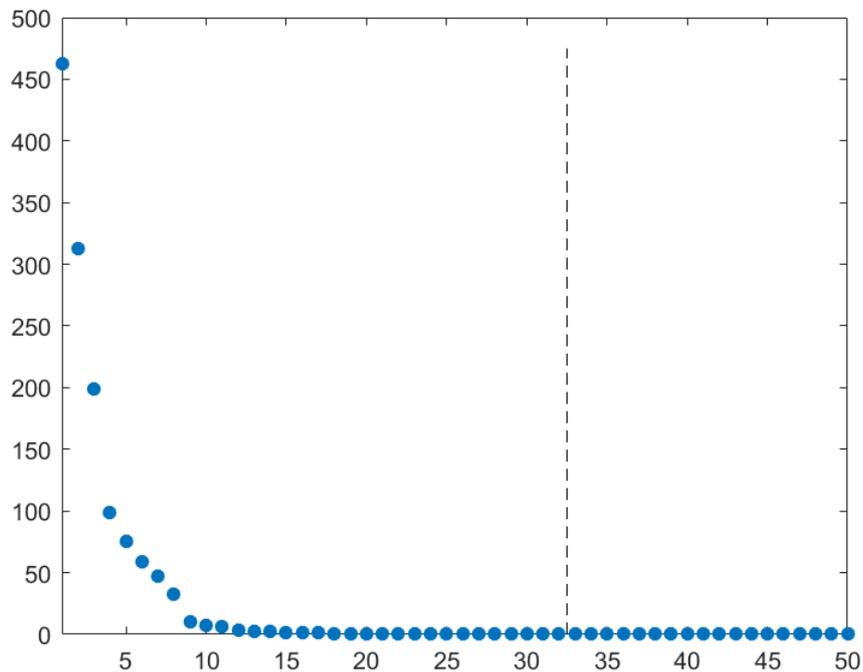


FIGURE 5.11: The 50 first eigenvalues from a mid-ventricular slice in a test image set. The dashed line marking the 20% of the values that were used in the model.

5.4.4 Applying the model on mid-ventricular slices

Since all input data from the user is set in a mid-ventricular slice, the model is first applied on this slice before being adjusted to the remaining slices.

Initially, \mathbf{b}_K is set to 0, making the first approximation of the segmentations equal to the final mean shapes from the alignment process; i.e. $\mathbf{u} = \bar{\mathbf{u}}$, in

consistency with Equation (4.12). The mid-ventricular segmentation, \mathbf{u}_{mid} , was then aligned to the edge map, \mathbf{v} , using procrustes analysis.

The mean shape was now roughly positioned close to the corresponding edge map, making it possible to compute the weights, \mathbf{b}_K , as in Equation (4.13). Creating an approximate segmentation, using Equation (4.12), applied the deformation modes in \mathbf{P}_K . This new rough segmentation was aligned to the edge map so new weights could be computed and the process was iterated until convergence was achieved.

5.4.5 Applying the model on remaining slices

The resulting \mathbf{R}^* , \mathbf{t}^* and s^* variables used for the alignment of the mid-ventricular slice, were saved so that they could be used for transforming the other mean shapes in the set, in order to give a rough starting segmentation.

The edge maps for the rest of the slices were created in a similar manner as explained in Section 5.4.2, although no manual seed points or insertion points were marked in these images. As a seed point the centroid of the transformed mean shapes was used. This point was found using the MATLAB function REGIONPROPS [14] on the binary mask of the mean shape. The intensity at the right ventricle insertion point in the arrival time image for the mid-ventricular slice was saved, so the same threshold could be used for the rest of the slices in the corresponding set. These two adjustments made it possible to create the rest of the needed edge maps, \mathbf{v}_j .

Then, for each shape, the weights were computed, as in Equation (4.13) where the transformation matrix was applied, coherent with Equation (4.12). To make sure the weights did not grow too fast, they were limited using Equation (4.14). The segmentations were then aligned before new weights were computed and the process was iterated until convergence.

5.5 Attempted approach

Once a satisfying 2D model was created, an attempt on a three dimensional, 3D, approach was conducted. Each set in the 3D case was treated as a coherent volume, making the segmentation process create a model for the complete set. The extraction and preparation of the data in the 3D case was the same as for the 2D case.

5.5.1 Principal component analysis

For the last part of the training phase in the 3D case, principal component analysis was performed to find the transformation matrix. The same process, as described in Section 4.2.2, was used. In this case the process took place in the $2NM$ -dimensional space, with M denoting the number of slices containing the right ventricle in the current test image set. For example, if a set was to depict the right ventricle in 12 of its slices, the space would

become $2 \cdot 80 \cdot 12 = 1920$ dimensional. The covariance matrix, \mathbf{S} , was computed so the transformation matrix could be found, in this case consisting of $2NM$ possible vectors to use as deformation modes.

Figure 5.12 shows an example of the first 50 eigenvalues for one image set in the test set. In the 3D case were different amounts of deformation vectors tried.

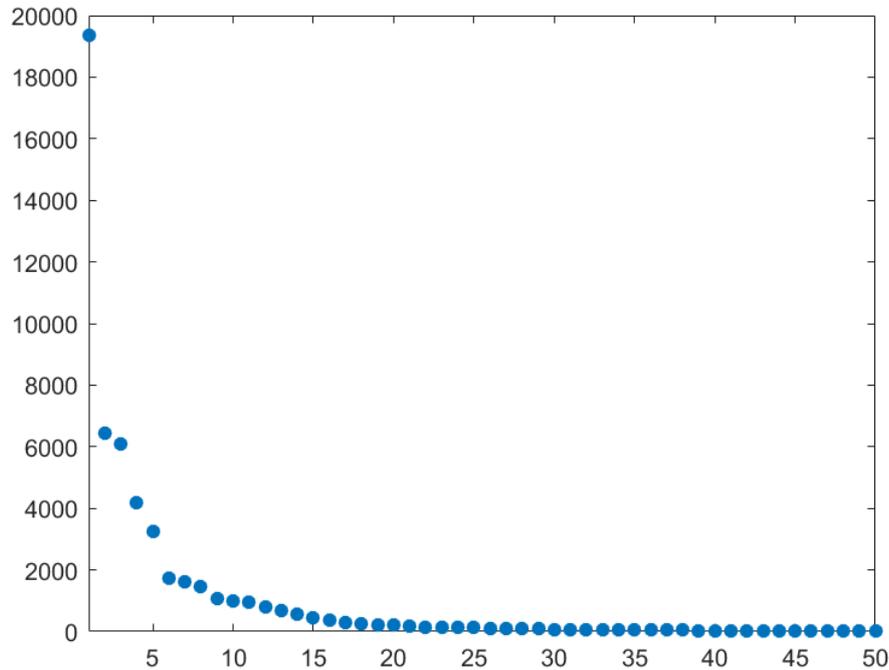


FIGURE 5.12: The 50 first eigenvalues for one of the test image sets.

5.5.2 Applying the model

For the first iteration, \mathbf{b}_K was initially set to 0, making $\mathbf{u} = \bar{\mathbf{u}}$, according to Equation (4.12). Since the user input was only made for the mid-ventricular slice, this was the only slice with an edge map thus far and the complete set could not be aligned, as one complete vector, yet. Since all slices from the set were in the same vector, all transformations had to be done at the same time. The transformation variables were therefore computed, using procrustes analysis, for the mid-ventricular slice. \mathbf{R}^* , \mathbf{t}^* and s^* were then applied on the complete mean shape set.

The remaining edge maps in the 3D case were found in the same way as in the 2D case. Since no manual insertion points were present in these images the intensity at the insertion point in the mid-ventricular arrival time image was saved to be used as a threshold for the rest of the slices. As a seed point the centroid in the corresponding transformed mean shape was used.

All edge maps were inserted into the same $NM \times 2$ -vector, \mathbf{v} , after which the distance between the edge maps and the mean shapes could be computed and the weights, \mathbf{b}_K , could be determined using Equation (4.13) and

limited using Equation (4.14). The model was then deformed, using Equation (4.12), and the process was iterated for the entire set until convergence was achieved.

5.6 Validation

The test data used for validation consisted of 23 image sets; 11 sets from patients and 12 sets from healthy volunteers. The right ventricle could be found in between 7 to 13 slices in these sets. Since the images from the transition area between the right ventricle and the pulmonary artery were difficult to use in the training data and therefore removed, these images were removed from the test data as well.

After the segmentation algorithm had been completed, the resulting segmentations were validated against the manual segmentations in all test sets. The results generated from the attempted 3D case had too poor quality and were not validated.

The physiological volumes of the segmented sets were computed, enabling comparison with the true volumes. This was done according to Equation (5.1), where M denotes number of slices, A is the area within the segmentation, r_x and r_y is the resolution in the x and y directions respectively, and t denotes the thickness of the slices.

$$V = \sum_i^M A_i r_x r_y t \quad (5.1)$$

Comparing volumes could be a misleading method, since it does not take into consideration if the segmentation is correctly located. Thus, the Dice-Sørensen coefficient was computed as well, by using Equation (4.15). This was performed on the complete volumes, as well as for only the mid-ventricular slices.

Chapter 6

Results

6.1 Example segmentations

Two examples of successful segmentations of mid-ventricular slices, can be seen in Figure 6.1 and 6.2; one from a healthy volunteer and one from a patient. Figure 6.3 shows another mid-ventricular slice from a patient, this one a bit less successful where surrounding fat has been included in the segmentation. Figure 6.4 shows two different slices from the same image set where the segmentation failed completely.

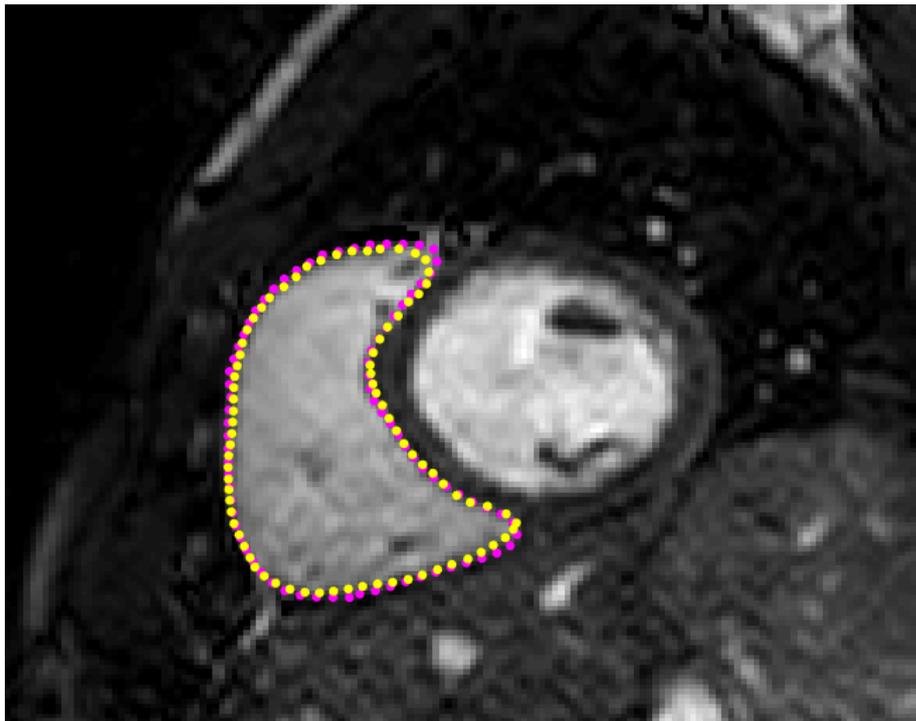


FIGURE 6.1: A successful segmentation of a mid-ventricular slice from a healthy volunteer's image set. The magenta coloured dots show the reference segmentation and the yellow dots show the automatic segmentation from the created model.

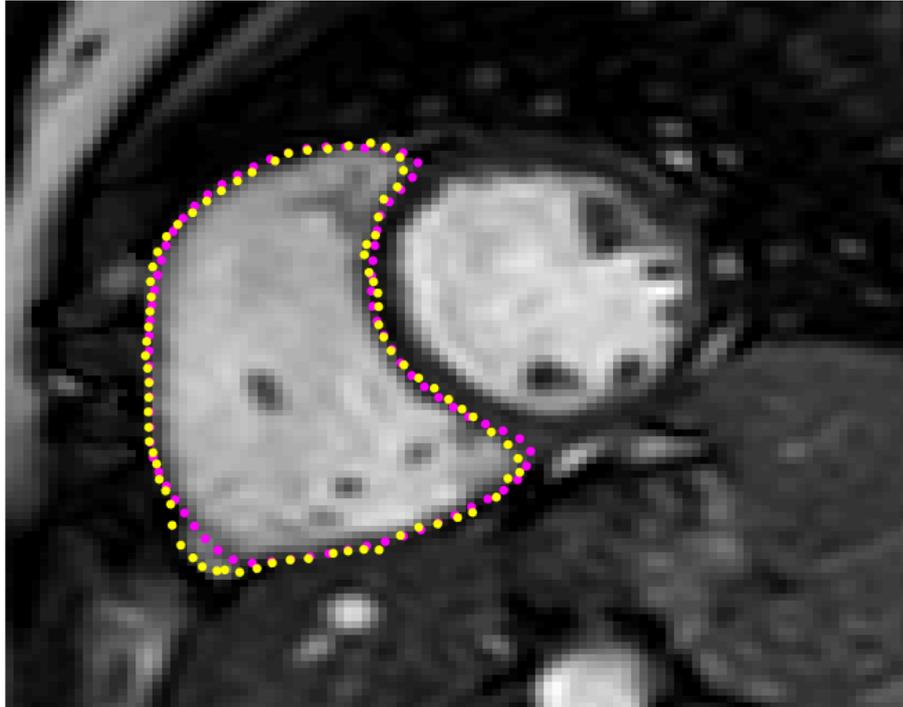


FIGURE 6.2: A successful segmentation of a mid-ventricular slice from a patient's image set. The magenta coloured dots show the reference segmentation and the yellow dots show the automatic segmentation from the created model.

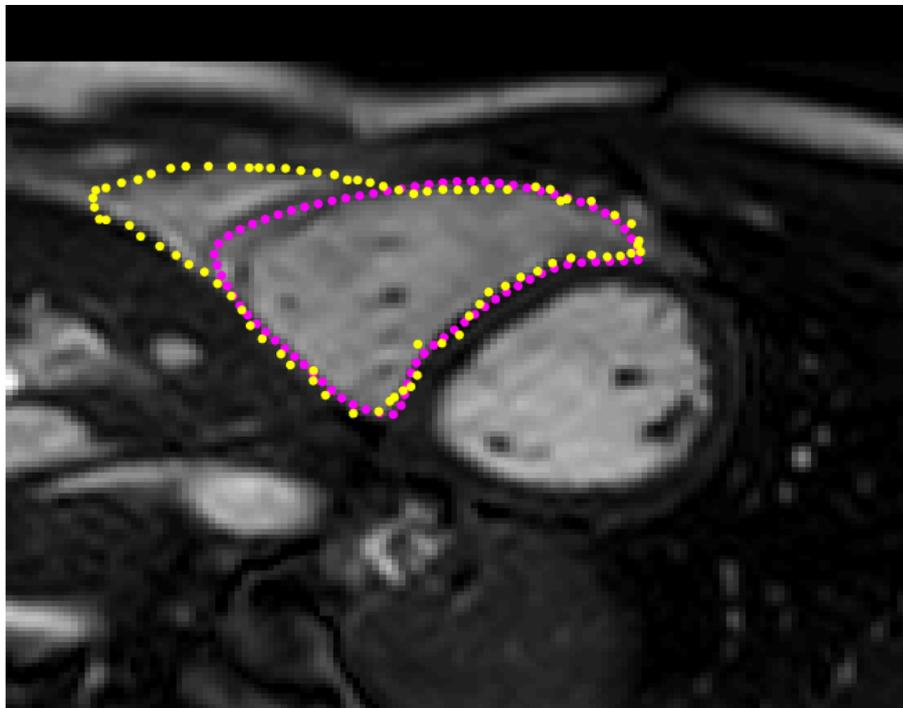


FIGURE 6.3: A less successful segmentation from another patient's set. The magenta coloured dots show the reference segmentation and the yellow dots show the automatic segmentation from the created model.

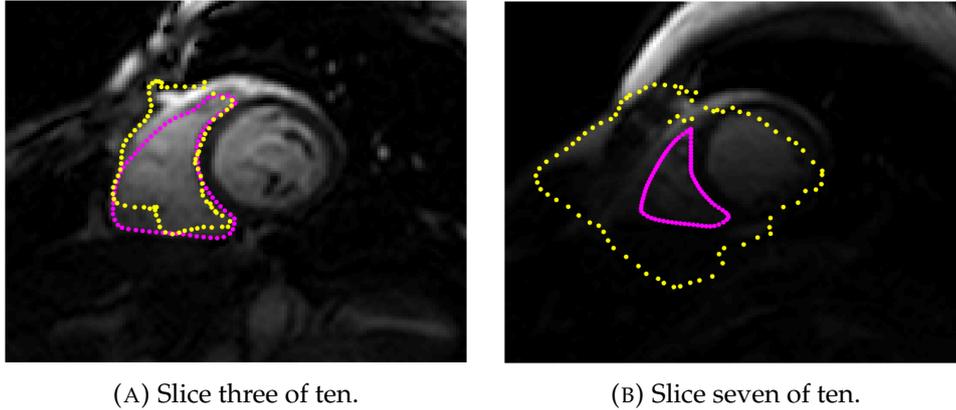


FIGURE 6.4: Two less successful segmentations from the same patient's image set. The magenta coloured dots show the reference segmentations and the yellow dots show the automatic segmentations from the created model.

The results for the 3D case were quite bad with a lot of incorrect boundaries, even though they managed to produce segmentations of reasonable size in most slices. An example of two segmentations can be seen in Figure 6.5, where the same slice was segmented using 20 and 160 deformation vectors respectively. Due to the discouraging results the 3D approach was not validated further.

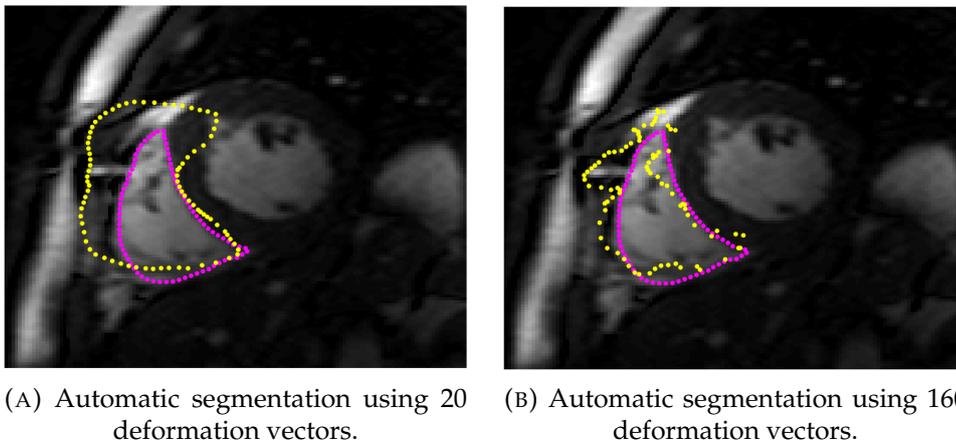


FIGURE 6.5: Two examples of segmentations of the same slice, using the 3D approach. The magenta coloured dots show the reference segmentations and the yellow dots show the automatic segmentations from the created model.

6.2 Validation results

The results from the volume computations and Dice-Sørensen coefficients can be seen in Table 6.1.

The volume comparisons can be seen in Figure 6.6, where the agreement is reasonable even though there were a few outliers. The correlation coefficient, R , was computed to approximately 0.39. The differences between the

Subject	Reference volume (ml)	Segmented volume (ml)	DS for complete set (%)	DS for mid-ventricular slice (%)
1	116.6	484	34	78
2	197.2	260	82	92
3	112.8	108.1	80	80
4	124.7	129.8	91	95
5	216.9	317.2	68	89
6	60.1	147.4	52	91
7	122.5	154.1	82	88
8	162	205.7	79	88
9	168.5	180.3	94	97
10	194.8	306.2	78	81
11	171.4	474.5	50	84
1	130.1	179.1	84	86
2	187	186.7	95	97
3	177.4	192.7	91	94
4	235.9	304.1	86	92
5	168.2	212.4	81	94
6	220	269.3	89	92
7	143.9	157.8	93	96
8	107.3	153.9	78	92
9	155.8	187.4	90	94
10	132.7	159.7	85	96
11	181.5	202.8	92	96
12	133.7	183.1	83	90

TABLE 6.1: Results from the validation of the test data. Blue area representing the patient sets while the red area represents the healthy volunteers.

true volumes and the segmented volumes were also compared and plotted in a Bland-Altman plot, see Figure 6.7, where a bias is noticed. The bias obtained a value and variability of 67 ± 90 .

The Dice-Sørensen coefficients for all total 229 slices from all sets, can be seen in Figure 6.8. The coefficients for the complete volumes, as well as for solely the mid-ventricular slices, can be seen in Figure 6.9. In Figure 6.10 can the Dice-Sørensen coefficients for the most basal and the most apical slices be seen respectively.

In all plots the results from patients are marked with blue circles and the results from healthy volunteers marked with red crosses.

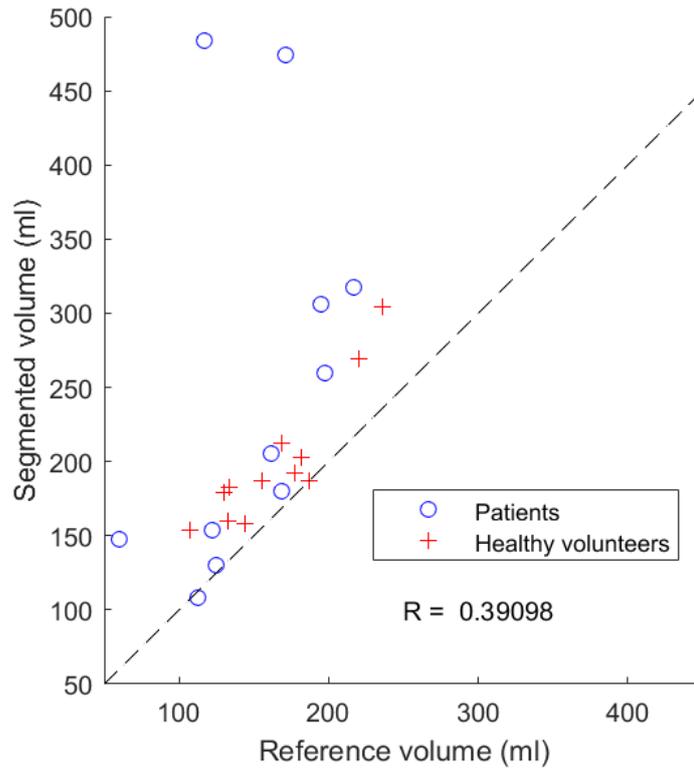


FIGURE 6.6: The segmented volumes compared to the reference volumes. The dashed line indicates the line of identity.

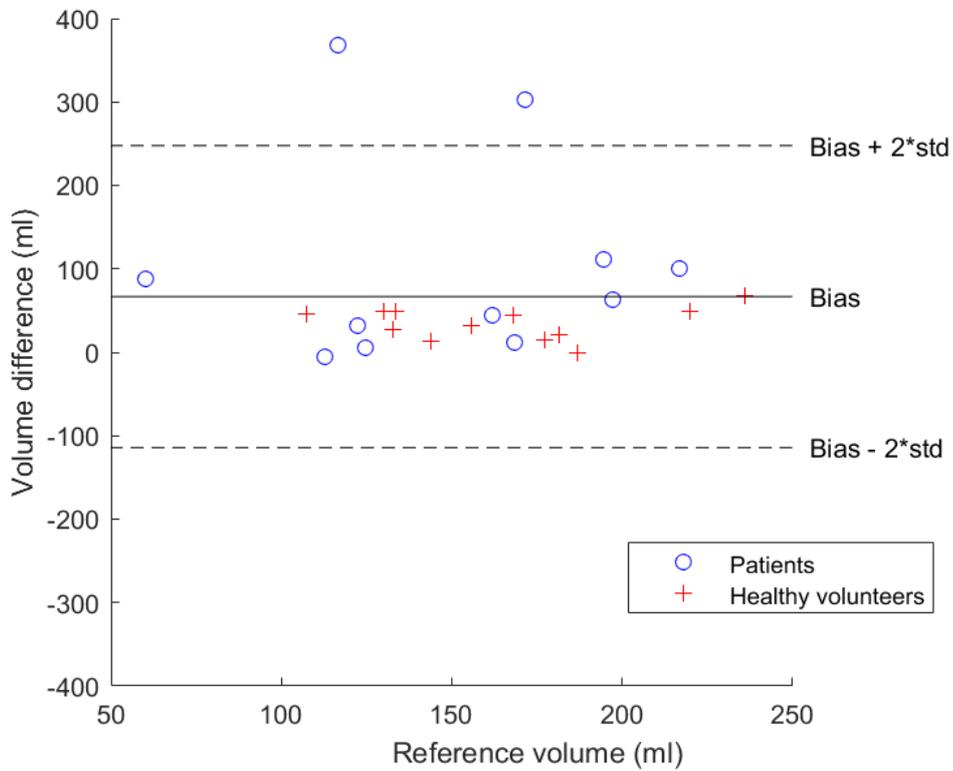


FIGURE 6.7: The differences between the reference volumes and the segmented volumes plotted as a function of the reference volumes.

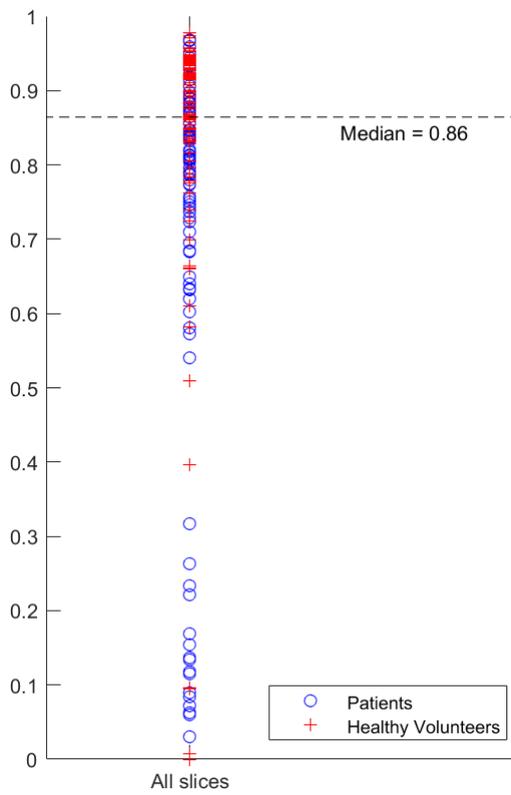


FIGURE 6.8: The Dice-Sørensen coefficients for all slices, in all sets, plotted. The dashed line represents the median value.

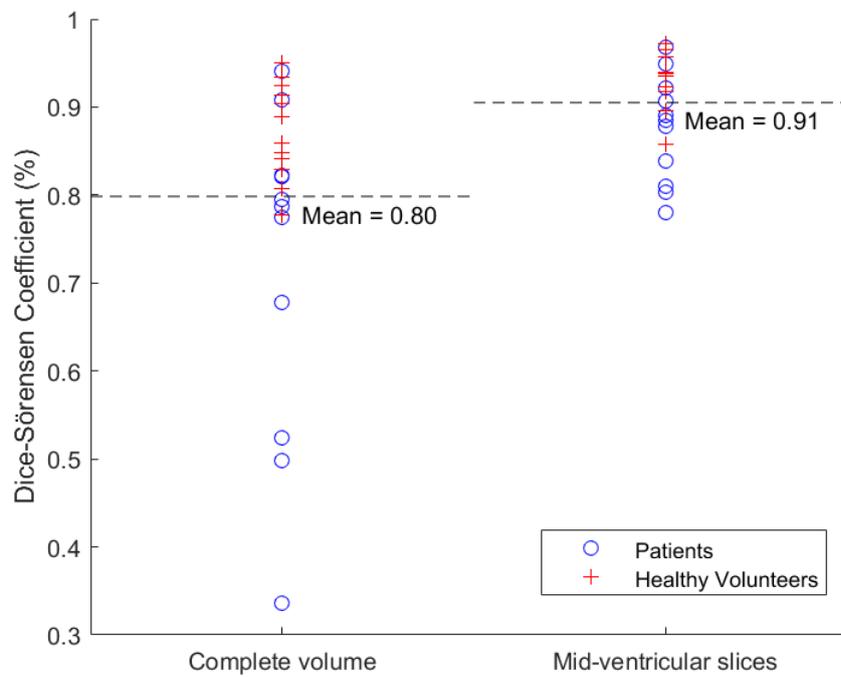


FIGURE 6.9: The Dice-Sørensen coefficients plotted, both for the complete volume as well as solely the mid-ventricular slices. The dashed lines denote the mean values.

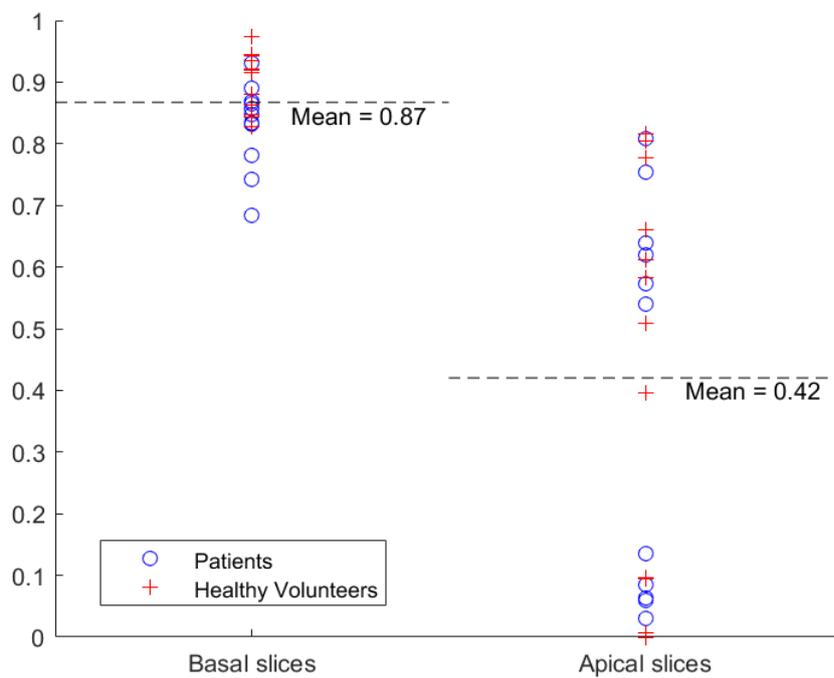


FIGURE 6.10: The Dice-Sørensen coefficients plotted for the most basal slices and the most apical slices. The dashed lines denote the mean values.

Chapter 7

Discussion and conclusion

In this thesis a segmentation method for the right ventricle in MR images has been presented. The algorithm was based on the active shape model and developed using the Segment software. The achieved results were generally good, except for a few outliers. The results will be discussed further in the following chapter before conclusions are drawn.

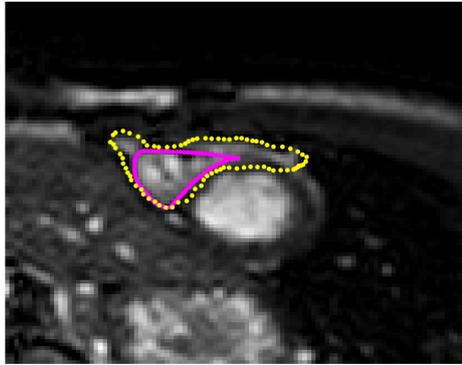
7.1 Discussion

The mid-ventricular slices, in which all inputs from the user were made, generally acquired good segmentations, which can be supported by the results in the right column in Figure 6.9. The Dice-Sørensen coefficients for these slices had a minimum value of 0.78 and a maximum of 0.97, indicating satisfying segmentations. The intensity threshold for each data set was computed from the mid-ventricular slices, which together with the manually placed seed points and right ventricle insertion points, aided the creation of accurate edge maps. This helped the model with alignment and deformation, leading to good segmentations of desired shapes in these slices, thus giving good results.

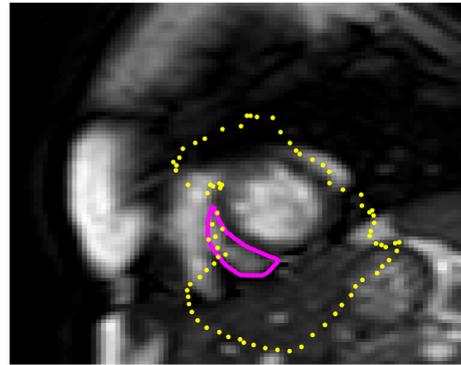
The closest slices to the mid-ventricular slice in each data set, did not differ too much in shape and size, and did therefore achieve good segmentations as well. The further away from the mid-ventricular slice in each set, the more the shapes changed in size, position and structure and they became harder to segment. The shape model mostly created very rough segmentations for the apical slices, like the examples in Figure 7.1. The left image shows an example where the model found different edges than the correct ones in the process. In a few apical slices the model could not create a segmentation at all, as in Figure 7.1b. In these images the contrast between the myocardium and surrounding tissue was too low, and the model could not find any edges at all. The Dice-Sørensen coefficients for the apical slices can be seen to the right in Figure 6.10. It is noticed in the plot that a group of sets have very low values and these represent the failed segmentations.

The basal slices in the sets did not differ too much in shape and intensity from the mid-ventricular slices and thus got fairly good segmentations as well. In Figure 7.2a segmentations are shown with slight jagged edges. Apart from the jagged edges, the algorithm did manage to find the correct shape and position. In Figure 7.2b another example of a basal segmentation is shown, this one with smoother edges. However, another part of the

image, with a similar intensity got included in the segmentation. The Dice-Sørensen coefficients for the basal slices can be seen to the left in Figure 6.10, where the minimum value is observed at 68% and the maximum value is observed at 97%.



(A) The apical slice, from a healthy volunteer's image set, where the model found the wrong edges.

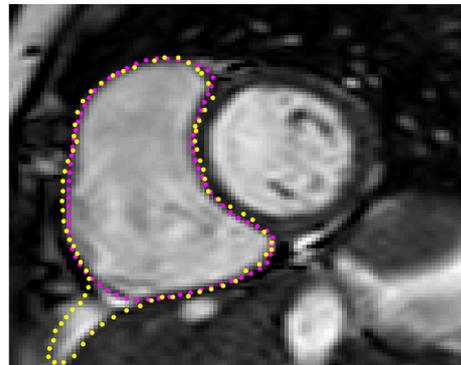


(B) The apical slice, from a patient's image set, where the model could not find any good edges at all.

FIGURE 7.1: Example segmentations of two apical slices from different image sets. The magenta coloured dots show the reference segmentations and the yellow dots show the automatic segmentations.



(A) The basal slice, from a healthy volunteer's image set, where the model created a segmentation with somewhat jagged edges.



(B) The basal slice, from a patient's image set, where the model created a good segmentation. However, an extra structure was falsely included in the segmentation, as seen in the bottom left corner of the image.

FIGURE 7.2: Example segmentations from two basal slices from different image sets. The magenta coloured dots show the reference segmentations and the yellow dots show the automatic segmentations.

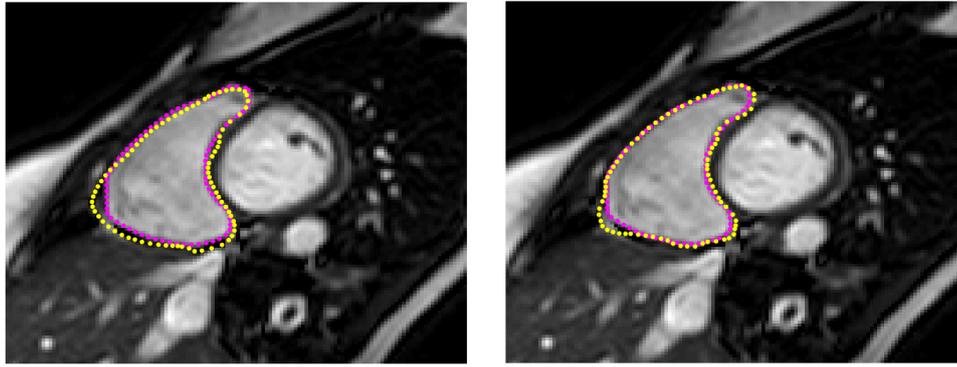
In Figure 6.8 the Dice-Sørensen coefficients from all slices can be observed. As can be seen, the coefficients of the resulting segmentations vary quite a lot, but the median at 86% is still sufficiently high. It can also be observed how most of the failed segmentations are from patient sets. The same can be noticed in the left side of Figure 6.9, where the Dice-Sørensen coefficients are plotted for the complete volumes. The mean from these results is 80%,

and the four sets with bad final segmentations shown in the plot did all come from patients. Possibly, the variational modes used in the model did not cover the deformations needed to create a patient's specific shape and thus bad segmentations were obtained.

Two examples of bad segmentations can be seen in Figure 6.3 and 6.4. In the first one the segmentation includes the right ventricle, as well as some surrounding fat. This segmentation defect is present in the entire set and arises from the edge shape, where the similarities between the intensity of fat and that of the right ventricle make it hard for the model to differentiate between the two structures. In the second image set the model is able to segment a few correct edges, as seen in Figure 6.4a, while it does not find any edges at all in Figure 6.4b. All slices in this set have very low intensity, making all edges very hard to detect, even with the naked eye. This makes it very difficult to create an edge map, thus leading to a bad final segmentation. Both of these data sets were from patients, contributing to the previously mentioned poorer results of the patients.

In Figure 6.6 volumes from the automatic segmentations are plotted as a function of the reference volumes. It is possible to see a linear dependence, even though almost every segmentation has a larger volume than its reference, with a bias at 67, see Figure 6.7. The two patient outliers are the two bad segmentations shown in Figure 6.3 and 6.4. The same two outliers can be seen in Figure 6.7.

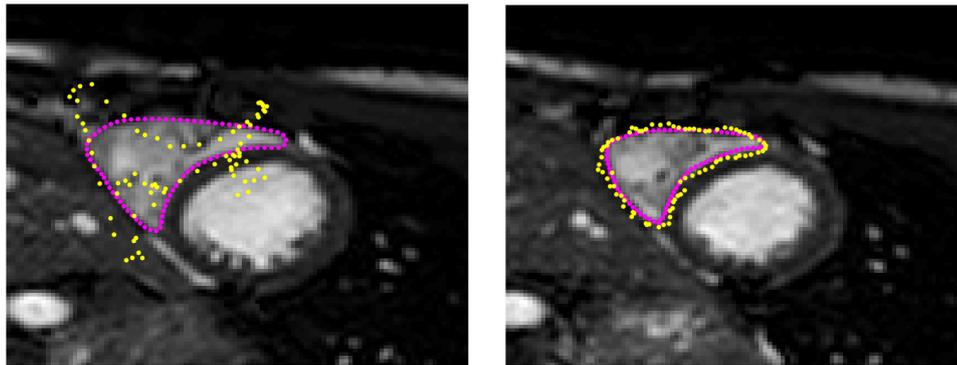
Observing Figure 5.11, it can be noticed how the eigenvalues approach 0 after about the 13th value. Different amounts of variational modes were tried, as seen in Figure 7.3. In the left image were 10 deformation vectors used, and in the right image were 20% of the vectors used, i.e. 32 vectors. The left image shows a satisfying segmentation whilst the right image shows a more detailed final segmentation. However, the different amounts of vectors were more noticeable in the more apical slices, due to their bigger variations in size and shape, see Figure 7.4. In the left image, the few variational modes do not really converge to the edge, as they manage to do in the right image. Due to the better results and since it was no noticeable difference in computational time, it was decided to use 20% of the vectors in the model.



(A) Automatic segmentation using 10 deformation vectors.

(B) Automatic segmentation using 32 deformation vectors.

FIGURE 7.3: Two examples of the same image, but with different amounts of variational modes. The magenta coloured dots show the reference segmentations and the yellow coloured dots show the automatic segmentations.



(A) Automatic segmentation using 10 deformation vectors.

(B) Automatic segmentation using 32 deformation vectors.

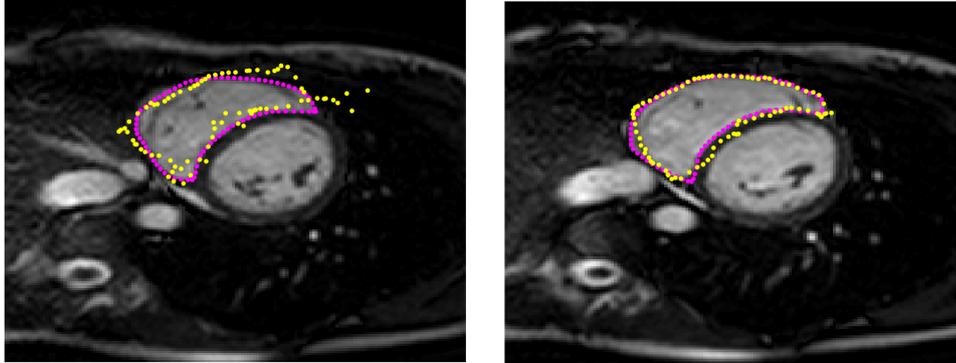
FIGURE 7.4: Two examples of the same apical image, but with different amounts of variational modes. The magenta coloured dots show the reference segmentations and the yellow coloured dots show the automatic segmentations.

7.2 Attempted 3D approach

The attempted 3D approach did not receive as much work as the 2D case, which may be an explanation for the discouraging results. The problem may be in the code or perhaps the proposed model. Intuitive about 20 of the deformation vectors would be used, when observing Figure 5.12. As seen in Figure 6.5a this would result in a smoother segmentation, but not as detailed. In Figure 6.5b the segmentation is better aligned, but with a much more jagged edge. Neither of these methods did converge to the edge map.

Since all slices in the same set will contribute when computing the weights, using all slices, as well as the five most central ones, was tested in order to compare the differences, see Figure 7.5. In both cases 10% of the deformation vectors were used. In Figure 7.5b the edge is a little less ragged than in

Figure 7.5a. A bad segmentation in one slice will contribute to big weights and thus give bad results in the other slices of the same set. Hence, this might not be the best approach before a better method for creating the edge maps has been implemented.



(A) All slices in the set was used to create the model.

(B) Only five mid-ventricular slices were used to create the model.

FIGURE 7.5: Two examples of segmentation using the 3D approach. The magenta coloured dots show the reference segmentations and the yellow coloured dots show the automatic segmentations.

7.3 Limitations and future work

A major limitation for the implementation of the active shape model is its dependence on the edge map. If the edge map fails, so does the model. Without a good edge map the active shape model cannot compute as accurate weights and the deformation will not be as desired. A possibility is to use a contrast image instead, where every pixel in the image can give information in which direction the model should look for the edge. Using this iterative process, no intensity information will be lost and new weights will be computed for a new edge map in every iteration.

The model cannot handle images with bad contrast, as seen in Figure 6.4, and it has difficulties separating between different tissues with similar intensities, as seen in Figure 6.3. These are two problems that might be solved with a better edge map as well.

Improving the edge detection technique might improve the segmentation of the apical slices as well. This could also be helped by relating the different slices to each other in some way. Currently, all slices are analysed separately and connecting them to each other, e.g. by looking at previous slices translations etc., could benefit the model.

Today, the user must give some inputs to be able to run the segmentation algorithm. For future implementations, a more user friendly interface would be to prefer. The results may even be a bit too independent of the inputs. Moving the right ventricle insertion point may give totally different results.

7.4 Conclusion

To conclude, an algorithm using the active shape model has been created to solve the difficult task of segmenting the right ventricle. The model used 92 subjects with manual delineations for training, and then validated the final algorithm on 23 test subjects, also with manual delineations. The algorithm gave good results for most image sets, but could not handle images with bad contrast.

The mean Dice-Sørensen coefficient of 80% for the complete volumes could be higher, and it has been discussed that this could be achieved by improving the process of creating edge maps. The image set with the best results gave a coefficient mean of 95% for the complete volume.

The suggested approach is promising, however the algorithm needs further improvements before final implementations. This model has only been adapted for the end-diastole state of the heart, and in the future, it would be beneficial to have an algorithm for the complete cardiac cycle. Different methods of improvements have been discussed and this algorithm is a good base for further development.

Bibliography

- [1] Tao Ai et al. "A historical overview of magnetic resonance imaging, focusing on technological innovations". In: *Investigative radiology* 47.12 (2012), pp. 725–741.
- [2] BruceBlau, Blausen Medical. *Midline Catheter*. [Online; accessed November 10, 2016]. URL: https://commons.wikimedia.org/wiki/File:Midline_Catheter.png.
- [3] Tim Cootes, E Baldock, and J Graham. "An introduction to active shape models". In: *Image processing and analysis* (2000), pp. 223–248.
- [4] Timothy F Cootes et al. "Active Shape Model - Their Training and Application". In: *Computer vision and image understanding* 61.1 (1995), pp. 38–59.
- [5] Lee R Dice. "Measures of the amount of ecologic association between species". In: *Ecology* 26.3 (1945), pp. 297–302.
- [6] RC Hawkes et al. "Nuclear magnetic resonance (NMR) tomography of the normal heart." In: *Journal of computer assisted tomography* 5.5 (1981), pp. 605–612.
- [7] Einar Heiberg et al. "Design and validation of Segment-freely available software for cardiovascular image analysis". In: *BMC medical imaging* 10.1 (2010), p. 1.
- [8] Huaifei Hu et al. "Automatic segmentation of the left ventricle in cardiac MRI using local binary fitting model and dynamic programming techniques". In: *PloS one* 9.12 (2014), e114760.
- [9] Jan Ainali. *MRI Philips*. [Online; accessed October 24, 2016]. URL: <https://commons.wikimedia.org/wiki/File:MRI-Philips.JPG>.
- [10] Elaine Nicpon Marieb and Katja Hoehn. *Human anatomy & physiology*. Pearson Education, 2007.
- [11] MathWorks, MATLAB User's Guide. *bwlabel*. [Online; accessed November 18, 2016]. URL: <https://se.mathworks.com/help/images/ref/bwlabel.html?searchHighlight=bwlabel>.
- [12] MathWorks, MATLAB User's Guide. *convhulln*. [Online; accessed November 18, 2016]. URL: <https://se.mathworks.com/help/matlab/ref/convhulln.html>.
- [13] MathWorks, MATLAB User's Guide. *graythresh*. [Online; accessed November 18, 2016]. URL: <https://se.mathworks.com/help/images/ref/graythresh.html?searchHighlight=graythresh>.
- [14] MathWorks, MATLAB User's Guide. *regionprops*. [Online; accessed November 18, 2016]. URL: <https://se.mathworks.com/help/images/ref/regionprops.html?searchHighlight=regionprops>.
- [15] Andriy Myronenko and Xubo Song. "Point set registration: Coherent point drift". In: *IEEE transactions on pattern analysis and machine intelligence* 32.12 (2010), pp. 2262–2275.

- [16] Németh László. *Extreme points*. [Online; accessed October 18, 2016]. URL: https://commons.wikimedia.org/wiki/File:Extreme_points.svg.
- [17] Norbert J Pelc et al. "Phase contrast cine magnetic resonance imaging." In: *Magnetic resonance quarterly* 7.4 (1991), pp. 229–254.
- [18] Caroline Petitjean et al. "Right ventricle segmentation from cardiac MRI: a collation study". In: *Medical image analysis* 19.1 (2015), pp. 187–202.
- [19] Hans H Schild. *MRI Made Easy:(-well Almost)*. Berlex Laboratories, 1992.
- [20] James A Sethian. "Fast marching methods". In: *SIAM review* 41.2 (1999), pp. 199–235.
- [21] Jasjit S Suri. "Computer vision, pattern recognition and image processing in left ventricle segmentation: The last 50 years". In: *Pattern Analysis & Applications* 3.3 (2000), pp. 209–242.
- [22] Bram Van Ginneken et al. "Active shape model segmentation with optimal features". In: *IEEE transactions on medical imaging* 21.8 (2002), pp. 924–933.
- [23] Weisstein, Eric W. *Vector Orientation*. [Online; accessed October 18, 2016]. URL: <http://mathworld.wolfram.com/VectorOrientation.html>.
- [24] Wikipedia, the free encyclopedia. *Diagram of the human heart*. [Online; accessed October 18, 2016]. URL: [https://commons.wikimedia.org/wiki/File:Diagram_of_the_human_heart_\(cropped\).svg](https://commons.wikimedia.org/wiki/File:Diagram_of_the_human_heart_(cropped).svg).