

# Registration and Analysis of Myocardial Perfusion Magnetic Resonance Images

Hildur Ólafsdóttir

February 2004  
Section for Image Analysis  
Informatics and Mathematical Modelling, IMM  
Technical University of Denmark, DTU



# Preface

This thesis is based on five months work at the section for Image Analysis, Informatics and Mathematical Modelling (IMM), Technical University of Denmark (DTU). The work accounts for 30 ECTS units and is a partial requirement for obtaining the degree Master of Science in Engineering (M.Sc.Eng).

Lyngby, February 2004



# Acknowledgements

First of all, I would like to thank my supervisor Mikkel B. Stegmann for all his good comments, ideas and suggestions throughout the project. My supervisor Rasmus Larsen is also thanked for taking the time to give feed-back and comments even though he was on sabbatical leave.

Hans Henrik Thodberg is thanked for kindly providing the basic implementation of MDL.

Henrik B. Larsson and the people at the MRI department at Hvidovre hospital are thanked for providing the data material. Without them, this kind of project would not have been possible.

I also want to thank everybody at the image analysis group, IMM. I could not think of better people to work with.

Finally, warm thanks to my boyfriend Thorhallur for his support and encouragement.



# Abstract

This thesis presents the registration and analysis of myocardial perfusion magnetic resonance images. Registration is carried out by a method based on Active Appearance Models (AAM). Extensions to the basic AAM include clustering of texture vectors in temporal dimension in addition to slice-coupled modelling.

A training set for the AAM is generated semi-automatically. This involves implementation of a tool to allow manual marking of points on object outlines and interpolation of the points by cubic splines to generate a full shape contour. Given the shape contours, point correspondences are obtained by Minimum Description Length (MDL) shape modelling. Requirements for the MDL approach imply that the cardiac shape cannot be optimised directly. Consequently, it is splitted into five contours at anatomical and pseudo landmarks and each contour set is optimised with influence from the full shape.

After registration of the set of images, perfusion assessment is carried out by providing pixel-wise signal intensity curves from the myocardium. Subsequently, perfusion maps are generated for three parameters derived from the curves: Maximum upslope, peak and time-to-peak.

Qualitative and quantitative validation is carried out on the presented methods. Data from 10 patients with acute myocardial infarction is provided for this purpose. MDL optimisations resulted in 10–14% improvement in terms of the description length and 10–28% improvement in terms of the total model variance. A leave-one-out cross validation was carried out for the registration method. This resulted in mean point to curve distance of  $1.25 \pm 0.36$  pixels. Perfusion maps for maximum upslope and time-to-peak positioned ischemic segments in most of the patients.

**Keywords:** Motion-compensation, registration, myocardial perfusion MRI, active appearance models, minimum description length, cubic splines, contour extraction tool, perfusion assessment, perfusion map



# Contents

<b>Contents</b>	<b>9</b>
<b>1 Introduction</b>	<b>13</b>
1.1 Motivation and objectives . . . . .	13
1.2 Thesis overview . . . . .	17
1.3 Publications during thesis work . . . . .	20
1.4 Terminology and synonyms . . . . .	20
1.5 List of Abbreviations . . . . .	21
1.6 Mathematical notation . . . . .	22
1.7 Nomenclature . . . . .	22
<b>2 MRI concepts and data material</b>	<b>25</b>
2.1 Basic concepts of MRI . . . . .	25
2.2 Myocardial perfusion MRI . . . . .	29
2.3 Data acquisition . . . . .	30
2.4 Summary and next steps . . . . .	32
<b>3 Related work</b>	<b>33</b>
3.1 Reported methods for registration of myocardial perfusion . . . . .	33
3.2 Summary and next steps . . . . .	36
<b>4 Preliminaries for statistical appearance modelling</b>	<b>37</b>
4.1 Definitions of shape and texture . . . . .	37
4.2 Landmarks . . . . .	38

4.3	Principal component analysis . . . . .	39
4.4	Procrustes alignment . . . . .	40
4.4.1	Planar ordinary Procrustes analysis . . . . .	41
4.4.2	Planar generalised Procrustes analysis . . . . .	41
4.5	Summary and next steps . . . . .	42
<b>5</b>	<b>Tool for manual extraction of shape contours from images</b>	<b>43</b>
5.1	Contour extraction tool and placing of marks . . . . .	43
5.2	Interpolation of marks by cubic splines . . . . .	45
5.3	Examples from the contour extraction tool . . . . .	48
5.4	Conclusions . . . . .	49
5.5	Summary and next steps . . . . .	50
<b>6</b>	<b>Minimum description length shape modelling</b>	<b>51</b>
6.1	Review of automatic model building approaches . . . . .	51
6.2	Basic concepts of MDL shape modelling . . . . .	52
6.2.1	Parameterisation function . . . . .	53
6.2.2	Objective function . . . . .	54
6.2.3	Minimisation of the description length . . . . .	60
6.2.4	Corpus callosum case . . . . .	61
6.3	Adjustments of cardiac shape contours to the MDL framework . . . . .	63
6.3.1	Modelling issues . . . . .	63
6.3.2	Splitting up cardiac shape contours . . . . .	64
6.3.3	Optimisation procedure . . . . .	65
6.3.4	Balanced alignment . . . . .	66
6.3.5	Computation time and implementation and issues . . . . .	66
6.3.6	Post processing . . . . .	67
6.3.7	Summary of cardiac MDL adjustments . . . . .	68
6.4	Summary and next steps . . . . .	69
<b>7</b>	<b>AAM based registration method</b>	<b>71</b>
7.1	Basic concepts of AAM . . . . .	71

7.2	Modelling of perfusion MRI time-series . . . . .	73
7.3	Adding cluster awareness . . . . .	75
7.4	Modelling object interfaces . . . . .	76
7.5	Multi-slice shape modelling . . . . .	76
7.6	Estimating and enforcing pose and shape priors . . . . .	77
7.7	Model initialisation . . . . .	79
7.8	Summary and next steps . . . . .	79
<b>8</b>	<b>Perfusion assessment</b>	<b>81</b>
8.1	Semi-quantitative perfusion assessment . . . . .	82
8.2	Summary and next steps . . . . .	84
<b>9</b>	<b>Results and evaluation</b>	<b>85</b>
9.1	Training set formation: Results from MDL . . . . .	85
9.1.1	MDL optimisation of the five sub-contour sets . . . . .	86
9.1.2	Full training set . . . . .	92
9.1.3	Discussion of MDL results . . . . .	93
9.2	Image registration by AAM . . . . .	94
9.2.1	Model training . . . . .	94
9.2.2	Cross validation of registration accuracy . . . . .	98
9.2.3	Discussion of registration results . . . . .	105
9.3	Perfusion assessment . . . . .	107
<b>10</b>	<b>Summary and conclusions</b>	<b>113</b>
10.1	Summary . . . . .	113
10.2	Conclusions . . . . .	114
<b>11</b>	<b>Future work</b>	<b>119</b>
	<b>Bibliography</b>	<b>121</b>
	<b>A Data file structure</b>	<b>127</b>

<b>B Annotation tool – instructions</b>	<b>129</b>
B.1 Help file for annotation tool . . . . .	129
<b>C Additional results</b>	<b>131</b>
C.1 MDL results, non-temporal set . . . . .	131
C.1.1 MDL Results for right epicardial contour set . . . . .	131
C.1.2 MDL results for left epicardial contour set . . . . .	136
C.1.3 MDL results for left endocardial contour set . . . . .	141
C.1.4 MDL results for right ventricle (RV) contour set . . . . .	146
C.2 Perfusion maps – AAM automatic registration . . . . .	151
C.2.1 Perfusion maps for patient 1 . . . . .	151
C.2.2 Perfusion maps for patient 2 . . . . .	152
C.2.3 Perfusion maps for patient 3 . . . . .	153
C.2.4 Perfusion maps for patient 4 . . . . .	154
C.2.5 Perfusion maps for patient 5 . . . . .	155
C.2.6 Perfusion maps for patient 6 . . . . .	156
C.2.7 Perfusion maps for patient 7 . . . . .	157
C.2.8 Perfusion maps for patient 8 . . . . .	158
C.2.9 Perfusion maps for patient 9 . . . . .	159
C.2.10 Perfusion maps for patient 10 . . . . .	160
C.3 Perfusion maps – Ground truth . . . . .	161
C.3.1 Perfusion maps for patient 1 . . . . .	161
C.3.2 Perfusion maps for patient 2 . . . . .	162
C.3.3 Perfusion maps for patient 3 . . . . .	163
C.3.4 Perfusion maps for patient 4 . . . . .	164
C.3.5 Perfusion maps for patient 5 . . . . .	165
C.3.6 Perfusion maps for patient 6 . . . . .	166
C.3.7 Perfusion maps for patient 7 . . . . .	167
C.3.8 Perfusion maps for patient 8 . . . . .	168
C.3.9 Perfusion maps for patient 9 . . . . .	169
C.3.10 Perfusion maps for patient 10 . . . . .	170

# Chapter 1

## Introduction

### 1.1 Motivation and objectives

Cardiovascular diseases (CVD)<sup>1</sup> are the leading cause of death in the world accounting for almost every third death or 16.6 million annually [4]. Since the population is aging, occurrences of CVD are increasing. Consequently this increases the need for accurate examination techniques. The traditional methods are echocardiography, X-ray angiography, nuclear medicine and ultrasound. Recently, Magnetic Resonance Imaging (MRI) has proven to be a powerful method for observing the heart because it does not have any side effects on the patient, it is flexible, and it gives highly detailed images compared to the alternative methods. In order to fully utilise this, accurate image analysis techniques are needed.

The subject of this study is myocardial perfusion MRI. In this type of MRI, a paramagnetic contrast agent is injected to the patient and multi-slice, image sequences over time are generated providing a 4D magnetic resonance (MR) myocardial perfusion data.

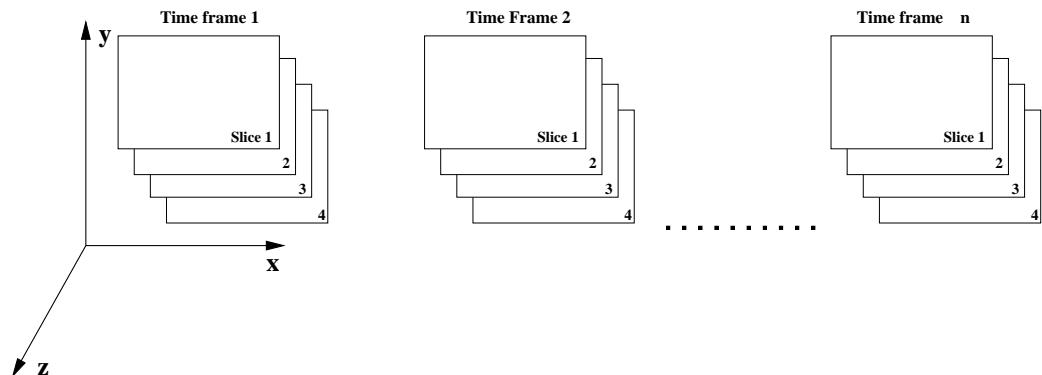
To allow the reader to become familiar with the data and its relation to regions of the heart, a brief introduction will be given here. This is only intended to give an overview, a more thorough discussion of the data and its properties is given in Chapter 2.

The above mentioned 4D structure is shown in Figure 1.1. The data consists of a 2D image frame in the  $x$ - $y$  plane for each slice in the  $z$  plane. This type of slice set is provided for each time frame in the series ( $t$ ).

Each image frame of the data produces a cross-sectional view of the patients chest wall. Figure 1.2 shows one multi-slice time frame including four consecutive slices

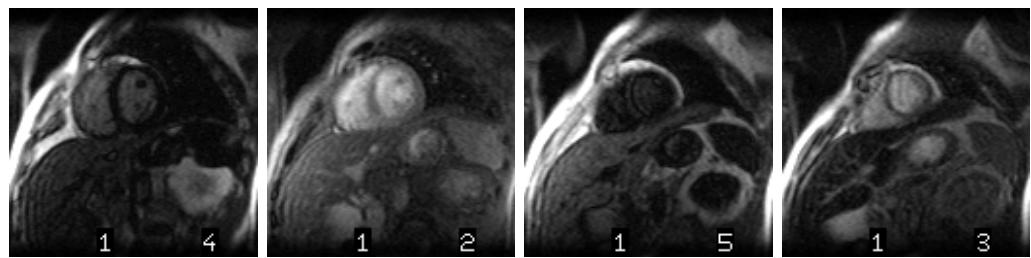
---

<sup>1</sup>CVDs are diseases which affect the proper functioning of the heart and blood vessels, the most important ones are: Myocardial infarction (heart attack), cerebrovascular diseases (stroke), coronary artery disease, transient ischemic attacks (TIA) and peripheral vascular diseases.



**Figure 1.1:** Dimensional structure of myocardial perfusion data. Four slices of 2D image frames shown in a time series.

in *basal* (upper end) to *apex* (lower tip of the heart) direction from left to right.

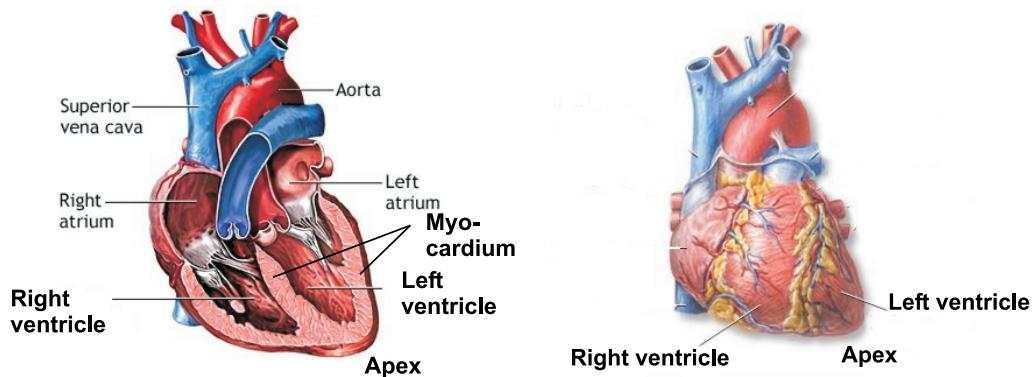


**Figure 1.2:** MR myocardial perfusion data: One four-slice time frame of the cardiac data. In basal-apex direction from left to right: Slice 1–4. Ignore the digits on the images at this point.

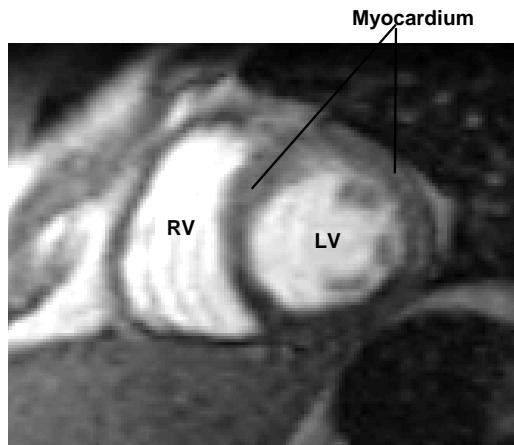
The regions of the heart which are of interest in this project are the *right ventricle* (RV), *left ventricle* (LV) and *myocardium* (heart muscle). To relate those regions to the data given in Figure 1.2, Figures 1.3 and 1.4 are provided.

Figure 1.3 shows two schematic views at the heart, the left one is a section through the middle and the right one is at the front.

For comparison, Figure 1.4 shows one slice after zooming in on the heart including labelling of the RV, LV and myocardium.



**Figure 1.3:** Schematic figures of the heart. Left: Section through the middle, the four chambers of the heart through which the blood flows. Right: Front view of the heart with right and left ventricles labelled. By courtesy of All refer health [3].



**Figure 1.4:** Myocardial MR data: An example image frame of the second slice zoomed in at the heart. RV, LV and myocardium are labelled.

Now that a brief overview of the data has been given, the actual purpose of myocardial perfusion MRI can be addressed. The contrast agent injected into the patient passes through the different regions of the heart resulting in varying intensities in the images across the time frames. From those intensity differences in the image sequences, blood flow in the myocardium can be observed. This can lead to diagnosis of e.g. *coronary artery disease* or *ischemic heart disease*. This disease causes narrowing of the coronary arteries leading to reduced blood flow to the heart. In serious cases this leads to *myocardial infarction* (heart attack). Regions with reduced blood flow are known as *ischemic regions* and can be detected in the images by regions darker than the surroundings, due to reduced flow of the contrast agent. The ultimate goal of this type of examination is to provide absolute quantification of the blood flow. This leads to a measure comparable between different studies and patients and more accurate diagnosis. Before the

*perfusion assessment* (quantification of blood flow) can be carried out, the sequence of images must be registered so that intensity in corresponding locations in the myocardium can be assessed. This is typically a manual procedure which consists of marking up points of correspondence in the myocardium.

In this study, a method based on Active Appearance Models (AAMs) is used to replace this tedious and time consuming manual process by providing an automatic registration of the image sequences. This is a learning-based method, in which the learning phase consists of a statistical analysis of shape and intensities from a training set of images. New, valid instances of the class of images can then be synthesised by "legal" deformations of the training set. Modifications from the standard AAMs needed for the myocardial perfusion data consist of multi-slice coupling and texture clustering in the temporal dimension.

Formation of a training set is a crucial step in the model building phase. A training set consists of several images of the class being analysed. Each image in the training set is processed by annotating points on the desired object. The key property of the annotation is that the points should be placed at corresponding locations across the set of images. This is often carried out manually but recent developments have made this procedure semi-automatic. Here, a minimum description length (MDL) shape modelling is applied. This method automatically estimates the optimal point correspondences given the outline of the object extracted from the images.

The part of extracting the object's outline remains manual and for this purpose, a contour extraction tool is implemented so that arbitrary points on the outline can be defined. Subsequently the points are interpolated by a cubic spline to give the full object contour.

The above mentioned methods are evaluated on four-slice MR myocardial perfusion sequences from 10 patients suffering from myocardial infarction.

After registration of the image sequences, a semi-quantitative myocardial perfusion assessment is carried out providing a basis for a full quantification of the blood flow.

To summarise, the main objectives of this project are to:

- Develop a tool for manual extraction of shape contours from myocardial perfusion data.
- Use MDL shape modelling to automatically generate landmarks of optimal correspondence on the extracted contours.
- Register the multi-slice image sequences using an AAM-based method.
- Perform semi-quantitative perfusion assessment on the registered image sequences.

## 1.2 Thesis overview

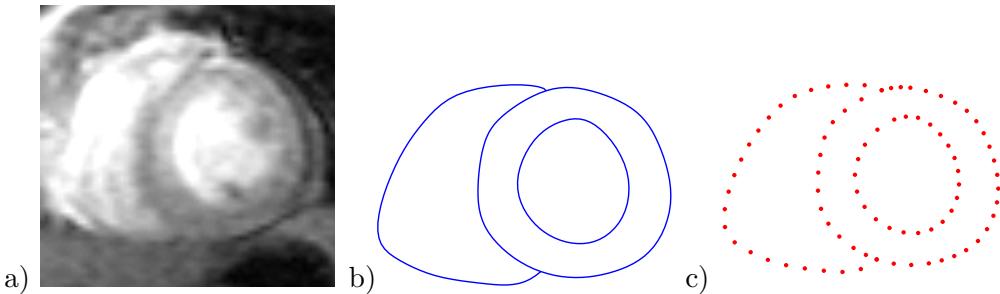
The thesis is structured as follows:

- **Chapter 2** covers basic concepts of MRI, particularly myocardial perfusion MRI along with the acquisition of data for this study.
- **Chapter 3** gives an overview of reported automatic registration methods for MR myocardial perfusion data.
- **Chapter 4** gives preliminary definitions for statistical appearance modelling, overview of principal components analysis (PCA) and Procrustes analysis. This chapter can be skipped by the reader familiar with those basic concepts.
- **Chapter 5** describes the tool implemented to ease manual extraction of the cardiac shape contours from the image frames.
- **Chapter 6** covers automatic placement of points of correspondence (landmarks) by MDL shape modelling along with modifications to the MDL framework to fit the myocardial perfusion data.
- **Chapter 7** gives a discussion of basic AAMs and the features added to model the myocardial data along with techniques to fit the model to unseen instances.
- **Chapter 8** discusses perfusion assessment based on registered image sequences.
- **Chapter 9** illustrates experimental results from MDL shape modelling, registration by the AAM based method along with example results for perfusion assessment.
- **Chapter 10** provides a summary, discussion and final conclusions.
- **Chapter 11** gives proposals for future work.

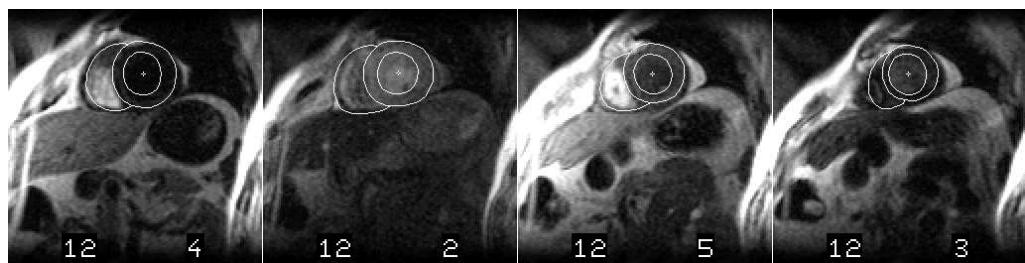
- **Appendices** include data file structure, instructions for the contour extraction tool and additional results not reported in Chapter 9.

To illustrate the procedure at hand, i.e. the development from contour extraction from raw images to perfusion assessment, Figures 1.5–1.7 are provided.

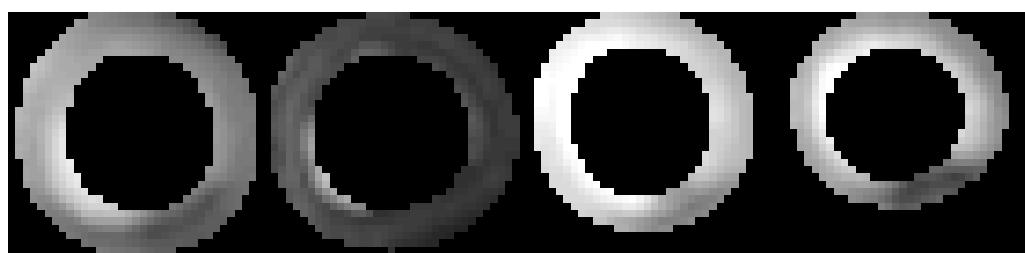
As may be understood from above, the project involves many different advanced techniques. Some of these techniques have been implemented by other researchers and their code is applied here, directly or with modifications. The methods are therefore not described in full detail in this thesis, but it is attempted to give the relevant reference where it applies.



**Figure 1.5: Step 1. Training set generation – placing of landmarks:** a) An image frame zoomed in at the myocardium. b) Outline of shape extracted by the contour extraction tool (Chapter 5). c) Automatically generated landmarks by MDL shape modelling (Chapter 6).



**Figure 1.6: Step 2. Registration:** Myocardium and RV detected by the AAM based registration method in an unseen four-slice image frame (Chapter 7).



**Figure 1.7: Step 3. Perfusion assessment:** Perfusion maps of the myocardium generated from a registered sequence of four slices from one patient. (Chapter 8). The maps are intended to locate areas with reduced blood flow.

### 1.3 Publications during thesis work

Parts of this thesis have been reported in the following paper:

M. B. Stegmann, H. Olafsdottir and H. B.W. Larsson. Unsupervised motion-compensation of multi-slice cardiac perfusion MRI. Invited contribution for the FIMH special issue in Medical Image Analysis, 2004 (submitted).

Work on MDL shape modelling related to parts of this thesis was published in the paper:

H.H. Thodberg, H. Olafsdottir. Adding Curvature to Minimum Description Length Shape Models, Proceedings of the British Machine Vision Conference, 2003.

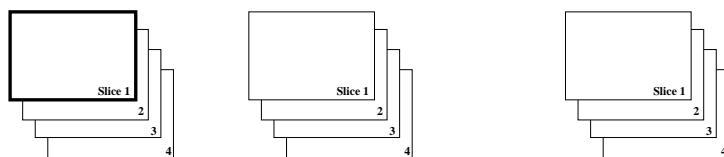
### 1.4 Terminology and synonyms

To ease reading and understanding, the terminology and possible synonyms used in this thesis sorted into different areas are provided.

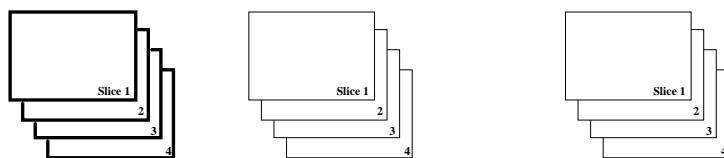
#### Data in general

Since the discussion of the 4D data can easily become confusing, the terminology will be related to the dimensionality structure in Figure 1.1. The regions in question are denoted by bold lines in the following.

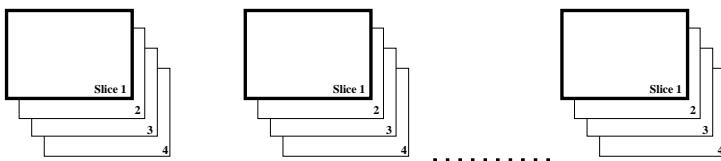
**Image frame, time frame, frame:** One 2D frame of one slice, one patient.



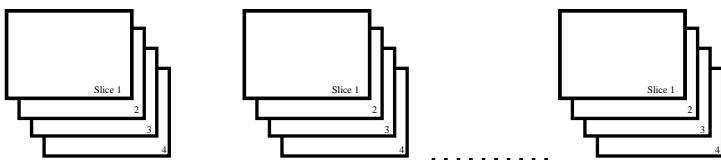
**Multi-slice frame, multi-slice time frame:** 2D frames of all slices, one patient.



**Slice sequence, perfusion sequence, sequence:** All time frames of one slice, one patient.



**Multi-slice image sequence, multi-slice sequence multi-slice perfusion sequence:** Full data for one patient, all slices, all time frames.



## Landmarking

**Cardiac shape, full cardiac shape** The outlines of RV, epicardium and endocardium extracted from an image frame.

**Full set:** The set of cardiac shapes extracted from the multi-slice sequence of all patients.

**Non-temporal set:** The set of cardiac shapes extracted from one multi-slice frame for all patients.

**Sub-contour:** A part of the cardiac shape, either RV, right epicardium, left epicardium, right endocardium or right endocardium (explained in detail in Chapter 6).

**Sub-contour set, sub-contour type:** Set of sub-contours.

**Local sub-contour set:** The sub-contour set being optimised.

## 1.5 List of Abbreviations

**MR:** Magnetic Resonance

**MRI:** Magnetic Resonance Imaging

**PCA** Principal Components Analysis

**DL** Description Length

**MDL** Minimum Description Length

**AAM** Active Appearance Model

**CAAM** Cluster-aware Active Appearance Model

## 1.6 Mathematical notation

The mathematical notation used in this thesis is listed below.

**Vectors** are formatted in columns and typeset in non-italic, lower-case, boldface using spaces to separate elements:  $\mathbf{v} = [ a \ b \ c ]^T$

**Matrices** are typeset in non-italic, boldface, capitals:

$$\mathbf{M} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

**Sets** are typeset using curly braces:  $\{\alpha \ \beta \ \gamma\}$  or  $\{\mathbf{x}_i\}_{i=1}^N$

**Vectors of ones of length  $N$**  are typeset as:  $\mathbf{1}_N$

**Identity matrices** are typeset as:

$$\mathbf{I} = \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}$$

**$p$ -th derivative of a function  $s(t)$**  is denoted as:  $s^{(p)}(t)$

## 1.7 Nomenclature

A list of symbols used in the thesis, divided into different areas, follows.

### MRI

$\omega_0$	Precession frequency
$\gamma$	Gyromagnetic ratio of an atom
$B_0$	Strength of external magnetic field
$M$	Net magnetisation
$\alpha$	The angle at which net magnetisation turns by
$T_1, T_2$	Relaxation times
$TI$	Inversion time

### Landmarks, shape representation

$\mathbf{s}_k$	A 2D shape vector for shape $k$
$\bar{\mathbf{s}}$	A 2D mean shape vector
$n_l$	Number of landmarks in a shape vector
$n_s$	Number of shapes in a set

## Procrustes analysis

$a, b$	Translation parameters
$\beta$	Scaling parameter
$\theta$	Rotation parameter
$\epsilon$	Error vector
$D^2$	Sum of squares errors
$\mu$	Full Procrustes mean shape
$\mathbf{z}_k$	$k$ -th standardised shape

## Splines

$s(t)$	Cubic spline
$t$	Curve parameter
$n$	Number of data points
$d_k, h_k$	Parameters for data point $k$ for moment representation of splines
$f(t)$	Underlying function describing behaviour of data
$\epsilon_k$	Error in data point $k$
$M(\mathbf{a}, t)$	Model with parameters $\mathbf{a}$ describing data in $t$

## Shape and texture modelling in general

$\Sigma_s$	Shape covariance matrix
$\phi_k$	$k$ -th eigenvector of $\Sigma_s$
$\Phi_s$	Matrix of shape eigenvectors
$\lambda_k$	$k$ -th eigenvalue of $\Sigma_s$
$\Lambda$	Diagonal matrix of eigenvalues
$\mathbf{b}_s$	Shape model parameters
$\mathbf{t}$	Texture vector
$\bar{\mathbf{t}}$	Mean texture vector
$\Phi_t$	Matrix of texture eigenvectors
$\mathbf{b}_t$	Texture model parameters
$V_T$	Total variance
$f_V$	Desired proportion of total variance to explain by a model
$n_m$	Number of modes

## MDL shape modelling

$\psi_k$	Parameterisation function of shape $k$
$Q$	Number of levels to base parameterisation function on
$\tau_{ij}$	Fractional distance of $j$ -th child node on level $i$ to its parents on level $i - 1$
$\mathcal{B}^m$	1D data set, the $m$ -th principal direction of all shape parameters
$\mathcal{L}^m$	Description length of principal direction $m$
$\mathcal{L}_{total}, DL$	Total description length
$\mathcal{P}$	Probability density function
$\Delta, \delta$	Accuracy in quantisation
$r, R$	Range of original data, range of data in shape space respectively
$\sigma_{min}, \sigma_{max}$	Minimum and maximum values of the standard deviation, $\sigma$
$n_g$	Number of principal directions where $\sigma > \sigma_{min}$
$n_{min}$	Number of principal directions where $\sigma \leq \sigma_{min}$
$\lambda_{cut}$	The smallest significant eigenvalue, equivalent to $\sigma_{min}^2$ .
$T$	Tolerance measure for stabilising MDL term
$N_k$	Number of landmarks used to describe sub-contour set $k$
$AL_k$	Average arclength of sub-contour set $k$

## AAMs

$\mathbf{W}_s$	Matrix of shape parameter weights
$\mathbf{c}$	Matrix of combined appearance model parameters
$\Phi_c$	Matrix of appearance eigenvectors
$\Phi_{c,s}$	Shape part of appearance eigenvectors
$\Phi_{c,t}$	Texture part of appearance eigenvectors
$\mathbf{p}$	Pose parameters warping object into image space
$\mathbf{q}$	Model parameters, a combination of appearance and pose parameters
$\mathbf{r}(\mathbf{q})$	Residual vector
$\mathbf{R}$	Parameter update matrix
$n_c$	Number of classes in texture clustering
$C$	Number of slices
$\kappa, \gamma, D_{max}$	Constants controlling influence of priors
$\Sigma_p$	Covariance matrix of sequence pose parameters
$\sigma$	Standard deviations of sequence shape parameters
$F_t$	$t$ -th frame
$S, P$	Number of first and last frame of stable period respectively

## Chapter 2

# MRI concepts and data material

This chapter provides a discussion of the basic concepts of MRI followed by the special case of myocardial perfusion MRI. Moreover, acquisition of data particularly for this study is described.

### 2.1 Basic concepts of MRI

Magnetic resonance imaging (MRI) has proven to be an invaluable tool for the medical world. It provides detailed images of the human body and has shown extreme flexibility with respect to different body parts, diseases, orientation, motion and so forth. Apart from these advantages, it is completely harmless for the patient unlike many existing methods. For the last few years, it has become increasingly widespread and in 2003 approximately 22.000 MRI scanners were used in 60 million examinations around the world [67]. Furthermore, in 2003 the nobel prize in physiology were awarded to the pioneers in development of MRI as it is to day, Paul C. Lauterbur and Peter Mansfield.

The remainder of this section is mainly based on Hornak [38], Skoog et al. [58] and Adeler [2].

The MRI technique is based on placing the patient inside a strong magnetic field (most current scanners operate at 1–3 Tesla). Atoms with odd number of protons or neutrons, for example hydrogen, sodium and phosphorus, possess a *spin angular momentum*. Due to its high abundance the hydrogen atom (proton) is the most widely used in MRI applications.

When the protons are placed in a strong *external magnetic field* as in the MR scanner, they act like compass needles due to their spin properties. This means

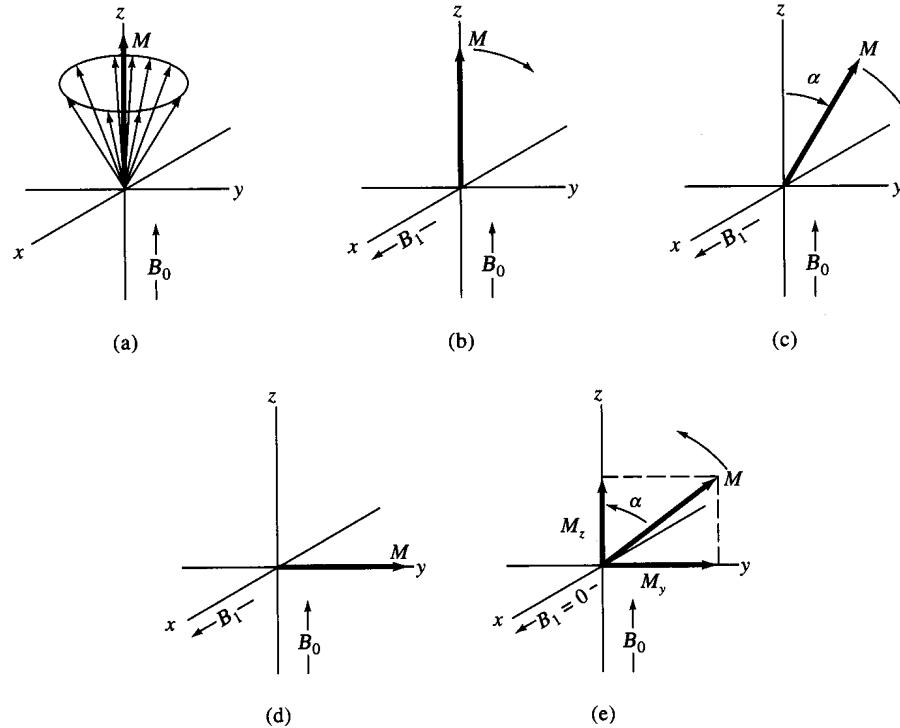
that some of them align themselves with the magnetic field, the lower energy level, and some of them align opposite to the magnetic field, the higher energy level.

Two spins in opposite direction cancel each other out but the remaining spins at the lower energy level, produce magnetic field along the direction of the external field called *longitudinal magnetisation*. The protons way to spin around the longitudinal axis is called *precession*. Further, they have a precession frequency  $\omega_0$ , which can be calculated by the Larmor equation given in Equation 2.1.

$$\omega_0 = \gamma B_0 \quad (2.1)$$

Here  $B_0$  is the strength of the external magnetic field and  $\gamma$  is the gyromagnetic ratio of the element being observed (here hydrogen).

The longitudinal magnetisation does not provide the signal needed to give information about a certain body part. By sending a *radio frequency (RF) pulse* through a coil of wire into the system, a second magnetic field is induced. This is referred to as *resonance* and allows some of the protons to move from a lower to a higher energy level, resulting in decrease in the longitudinal magnetisation. Other protons precess away from the longitudinal alignment into the transverse (x–y) plane, resulting in *transversal magnetisation*,  $B_1$ . This causes the net magnetisation,  $M$  to turn by an angle  $\alpha$ . After the RF pulse terminates, the protons relax, dephase in the transversal plane and reach equilibrium again. Figure 2.1 illustrates this in a rotating reference frame often useful when visualising the magnetisation.



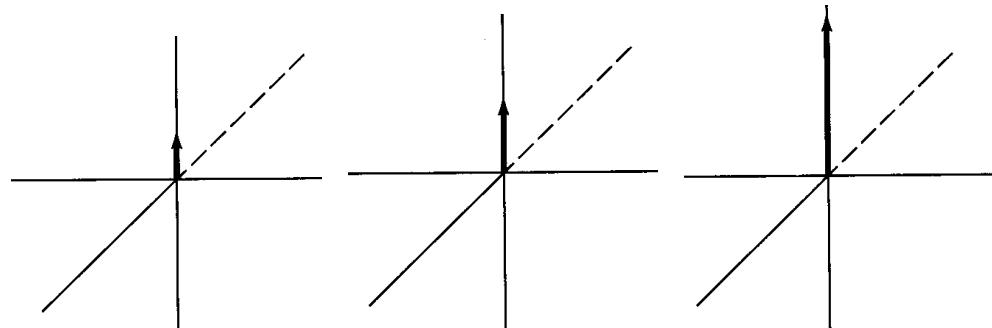
**Figure 2.1:** Behavior of magnetic moments of protons (each denoted by a vector in the cone) in a rotating reference frame (a) Before RF pulse. Net magnetisation ( $M$ ) is along the longitudinal axis. (b)-(d) 90° RF pulse is sent in the  $x$ -direction inducing a magnetic field,  $B_1$ . Rotation of the net magnetisation vector during lifetime of the pulse is illustrated. e) Relaxation after termination of the pulse. Net magnetisation turns back to the longitudinal direction. Longitudinal and transversal components of  $M$  ( $M_z$  and  $M_y$ ) are shown. By courtesy of Skoog et al. [58]

The rate of relaxation is assessed by two constants:

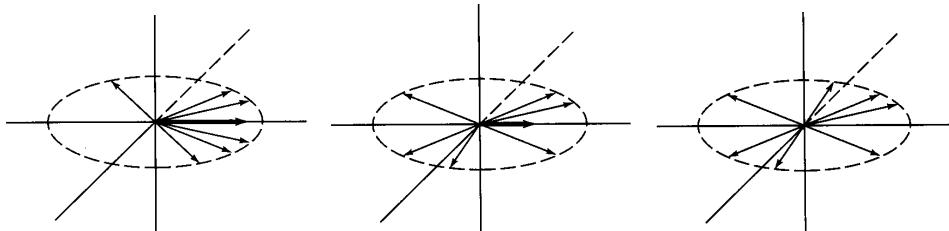
- $T_1$ : *Longitudinal relaxation time*, determines the time it takes to attain the previous longitudinal magnetisation.
- $T_2$ : *Transversal relaxation time*, determines the time it takes the transversal magnetisation to die out.

Figures 2.2 and 2.3 illustrate the longitudinal and transversal relaxation respectively.

Immediately after turning the RF pulse off, the protons emit a radio signal, the MR signal, of the same frequency as was sent into the system. Gradually the signal dies out as the system reaches equilibrium.



**Figure 2.2:** Longitudinal relaxation. Longitudinal magnetisation,  $M_z$  (bold vector) reaches previous strength in time determined by  $T_1$ . By courtesy of Skoog et al. [58]



**Figure 2.3:** Transversal relaxation. Transversal magnetisation,  $M_y$  (bold vector) dies out due to dephasing of magnetic moments in time determined by  $T_2$ . By courtesy of Skoog et al. [58]

Since the precession frequency of the protons is affected by the magnetic field strength by Equation 2.1, spatial variation of precession frequencies can be created by *gradient* fields which produce different strengths across the magnet and therefore different signals. Since the short duration of the radio signals, a sequence of RF pulses must be generated to obtain the sufficient information to generate an image from the observed signals.

The received MR signals depend on several factors. A few of them are listed below.

- Proton density and frequency
- $T_1$  and  $T_2$  relaxation times
- Metabolic properties
- Motion
- Flow and viscosity
- Contrast agents

In order to generate an image with good contrasts, the above mentioned factors are utilised by designing the pulse sequence so as to weight those factors differently. For example, water has shorter relaxation times than fat. This can be utilised by sending two radio pulses with appropriate time gap which optimises the relaxation difference between the tissues and thereby the image contrast. The resulting image

is referred to as a *T<sub>1</sub>-weighted* or *T<sub>2</sub>-weighted image* depending on which of the relaxation differences, longitudinal or transversal, are being optimised.

Paramagnetic contrast agents, for example gadolinium diethylenetriaminopen-taacetic acid (Gd-DTPA) shorten the relaxation times for the tissue it is exposed to, which given the appropriate pulse sequence can generate good contrasts in the image.

Commonly used pulse sequences are for example *Saturation recovery sequence*, which only uses 90° pulses ( $\alpha = 90^\circ$ ), *inversion recovery sequence*, which uses 180°pulse followed by a 90° pulse to generate T<sub>1</sub>-weighted images and a *spin echo sequence*, which uses a 90°pulse followed by 180°pulses.

Attempts have been made to make the imaging faster, for example FLASH (Fast low angle shot), which decreases the flip (pulse) angle below 90° to reduce the time between pulses.

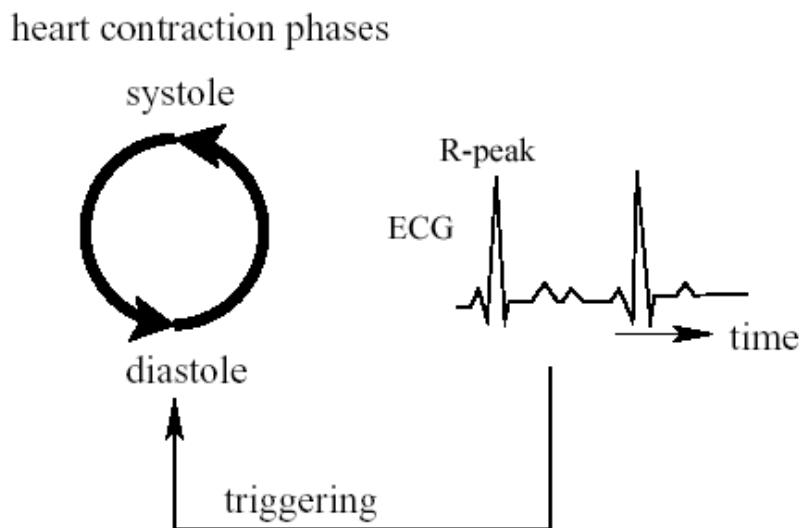
As stated above, the MRI process needs to be designed to fit the function of the tissue or organ being examined. As myocardial perfusion is the topic of this project, a brief discussion of the techniques and tricks applied to acquire this type of image sequences is given in the following section along with discussion of variation sources present in the images.

## 2.2 Myocardial perfusion MRI

Developments in MR-technology during the past decade have made it possible to acquire information about dynamic processes in the human body. As an example of such, myocardial perfusion imaging encompasses assessment of myocardial perfusion at rest and during stress (e.g. pharmacological). By injecting a bolus of a paramagnetic contrast substance the myocardial perfusion mechanism can be quantified. Thus, perfusion MR qualifies as an essential instrument in the assessment of ischemic heart diseases. As the contrast agent tags the blood stream and amplifies the MR signal by a shortening of the T<sub>1</sub> relaxation time, areas of the myocardium served by diseased arteries show a delayed and attenuated response.

Acquisition is carried out dynamically and registered to the heart cycle using electrocardiogram (ECG)-triggering. This means that motion of the heart is corrected by acquiring an image in the same position in each heart cycle. Figure 2.4 illustrates this. Imperfect ECG-triggering can give rise to depiction of erroneous heart-phases, in which tissue correspondences can be partially shattered, due to the long-axis movement of the left ventricle during the heart cycle.

Images are typically acquired from one or more short-axis slices every *n*-th heart-beat, where through-plane resolution is traded for temporal resolution and vice versa. If the acquisition time-window is sufficiently short (typically < 20 seconds), breath-hold can be used to remove respiration artifacts.



**Figure 2.4:** Simplified illustration of ECG triggering: Radio pulse is sent in the same time/position in each heart cycle corresponding to the heart contraction phases. Diastole: Both right and left ventricles are filled with blood. Systole: The blood leaves the right and left ventricles (and is pumped to the body). By courtesy of Spreeuwiers and Breeuwer [60].

In summary, the most prominent sources of variation in shape and appearance found in perfusion MRI (aside from general MR acquisition artifacts such as movements of the patient) include:

- Biological inter-subject variation
- Scan planning
- Contrast passage
- Contrast uptake
- Respiration
- Inaccuracies in ECG-triggering

## 2.3 Data acquisition

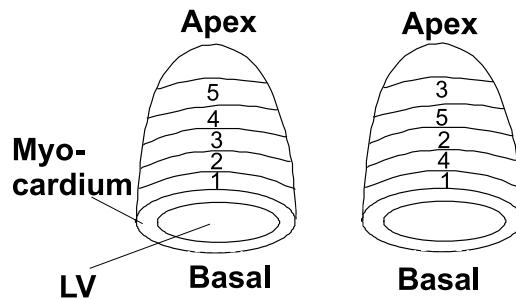
The data material provided for this study comprises 2500 myocardial perfusion, short-axis, magnetic resonance images (MRI) obtained from ten freely breathing patients with acute myocardial infarction. Five slices of 50 sequential image frames each were acquired before, during and after the bolus of contrast. The contrast agent was Gd-DTPA. Registration relative to the heart-cycle (end-diastole) was obtained using ECG-triggered acquisition from a whole-body Siemens Vision MR unit. Slice acquisition was restarted at every third R-peak, thus providing an approximate frame time of three seconds, depending on the heart rate. Further

details about the scanning procedure are given in Table 2.1.

**Table 2.1:** Scanning parameters

Field strength, $B_0$	1.5 Tesla
MR pulse sequence	Inversion recovery turbo-FLASH
Matrix size	$128 \times 128$
Field of view	$300 \times 300$ mm
Slice thickness	10 mm
Inter slice gap	0 mm
Storage bit depth	16 bit
Inversion times, TI	598 ms, 209 ms, 792 ms, 404 ms

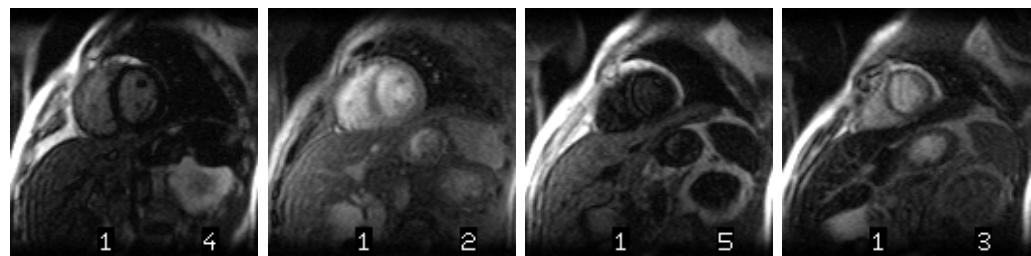
To avoid magnetic effects from adjacent slices, slices are not scanned in anatomical order as may be seen from the inversion times in Table 2.1. The relationship between the spatial order of the slices and the acquisition order is clarified in Figure 2.5.



**Figure 2.5:** Left: Anatomical order of slices. Right: Acquisition order of slices. Slices are acquired in this order to avoid magnetic effects from consecutive slices. Basal and apex are labelled to illustrate the hearts position.

Unfortunately, the first slice had to be discarded from the data set due to bad signal-to-noise ratio. The remaining four slices used in the study will be referred to in their anatomical order as **slice 1–4** (equivalent to slice 2–5 in Figure 2.5 left). An example multi-slice frame showing four slices in anatomical order (in basal–apex direction from left to right) from the first patient is given in Figure 2.6. The right digit marked on the images denotes the order of which the slices were acquired (as seen in the right of Figure 2.5). The left digit denotes the time frame number.

Figure 2.6 shows that there is not consistency in slice intensities, that is, the first and third slice appear darker on average than the remaining slices. This was observed for most of the data and can be explained by the inversion times, i.e. slices with longer inversion times give less signal and therefore appear darker. Those slices can be more difficult to register than the brighter ones due to poor signal-to-noise ratio.



**Figure 2.6:** Four slice images of patient 1 prior to bolus arrival (frame 1). Inversion times:  $\text{TI}=598 \text{ ms}$ ,  $209 \text{ ms}$ ,  $792 \text{ ms}$ ,  $404 \text{ ms}$ . Left digit denotes frame number, right digit denotes acquisition number of slice.

## 2.4 Summary and next steps

This chapter addressed the basic concepts of MRI, the special case of myocardial perfusion MRI and the acquisition and structure of data used in this study. Later in this thesis, details about how the multi-slice perfusion sequences are analysed and registered will be given. However, next chapter provides an overview of alternative registration approaches on similar data.

# Chapter 3

## Related work

In this chapter, the current status of registration of MR cardiac perfusion images as reported in the literature will be addressed.

Note that all the following approaches register a single slice sequence independent of the remaining slices from the scan unlike the simultaneous registration of multi-slice sequences presented in this project. Consequently, computation times provided in this chapter apply to registration of a single slice sequence.

### 3.1 Reported methods for registration of myocardial perfusion

Yang et al. [70] use phase difference between successive image frames to correct for translational motion. After tracing one frame from the sequence manually, the shape changes are compensated for by using a deformable model. They show example results from one patient where they compare SI-curves<sup>1</sup> before and after registration but lack further validation of the method.

Behloul et al. [6] simplify the problem of myocardial boundary detection by adding a slice from functional MRI to the perfusion sequence. They segment the functional slice by fuzzy clustering and fuzzy inference systems and warp the result to the perfusion sequence. The method requires manual definition of a rectangular region of interest (ROI) on each image of the perfusion sequence. The authors claim that their method can be applied in real-time but unfortunately it lacks validation.

Bidaut and Vallee [9] use a multi resolution translation/rotation based registration

---

<sup>1</sup>A signal-intensity (SI) curve is a plot of intensities in a region of the myocardial vs. time frame. From its properties, various parameters regarding myocardial perfusion can be calculated. Refer to Chapter 8.

minimising the mean squared differences (MSD) between each image frame and a reference frame. They evaluate their method using 8 sequences of 90 frames from 8 patients with stable coronary artery disease (CAD). The validation is based on a comparison of image correlation factors in addition to motion reduction measurements for manual (ground truth) and automatic registration. Their computation time is 5 minutes per 100 time frames. Dornier et al. [26] extend this work by defining cardiac masks with no signal intensity changes throughout the perfusion sequence. Computation time is reduced to 2.5 minutes per 100 time frames. Moreover, they improve the validation by calculating absolute myocardial perfusion and compare to values derived from manually registered sequences (ground truth).

Breeuwer and Spreeuwiers [15, 16, 60], also use a translation/rotation based registration but with normalised cross-correlation as a similarity measure. After the registration, an exact detection of endocardial and RV boundaries is obtained by region growing on feature images. The epicardial boundaries are detected by a deformable snake model. In [15] and [16] the method is evaluated on 36 perfusion sequences from 12 patients by comparing the calculated perfusion parameters to estimated values from X-ray angiograms of same patients. They obtain good correspondence for 11 of the patients. Their computation time is less than 4 minutes per slice sequence. In [60] qualitative evaluation on 30 perfusion sequences from 14 scans is carried out. They succeed in 26 out of 30 sequences and the computation time is 25 seconds per 70 time frames.

Gallippi and Gregg [34] present a statistics based registration method using deformable template matching. The similarity measure depends on local brightness variations and edge directions. They evaluate their method on 12 patients and measure its quality by means of motion reduction and comparison of signal-intensity (SI) curves prior to and after registration.

Ablitt et al. [1] use tissue tagging to correct for through-plane distortion on a multi-slice sequence. The in-plane motion in each slice is corrected for using free-form image registration with partial least squares regression (PLSR) deformation learning. They evaluate their method on a set of 8 patients with CAD and 5 healthy individuals in addition to a synthetic data set. The in-vivo data sets include 3 slices with 50 frames per slice. The accuracy of the method is measured relatively by comparing SI-curves before and after registration. Gao et al. [35] do further studies on the PLSR deformation modelling. Intrinsic pattern at the chest wall is used to correlate with deformation vectors at the myocardium, leading to a motion prediction approach. The method is evaluated using dataset from 9 patients with CAD, where data for each patient includes 3 slices and 50 frames per slice, as well as a synthetic dataset. The accuracy of the method is measured by comparing motion reduction for a free-form image registration versus the proposed method.

**Table 3.1:** Overview of registration methods for MR cardiac perfusion images.

Reference	Method	Similarity measure	Data	Validation method	Breathing Computation time	Fully automatic
[70]	Translation based registration, deformable model	–	1 patient (100 frames)	SI-curves	–	No
[6]	Segmentation of functional MRI mapped to perf. seq.	–	–	–	Real-time	No
[9]	Translation/rotation based registration	Mean squared difference	8 patients (8 sequences, 90 fr.)	Motion reduction, image correlation factors	5 min pr. 100 fr.	Yes
[26]	Translation/rotation based registration	Mean squared difference	8 patients (100 fr.)	Motion reduction, comparison of ground truth versus automatic SI-curves and quantitative perfusion parameters	2.5 min pr. 100 fr.	Yes
[15], [16]	Translation/rotation based registration, region growing, snake model	Normalised cross correlation	12 patients (36 seq., 60 fr.)	Quantitative comparison of perfusion parameters	4 min pr. seq.	Yes
[60]	Translation/rotation based registration, region growing, snake model	Normalised cross correlation	14 (30 seq., 70 fr.)	Visual inspection	25 sec pr. 70 fr.	Yes
[34]	Deformable template matching	Brightness, edge directions	12 patients	Motion reduction, SI-curves	–	Yes
[5]	Translation based template matching using edge and intensity information	Mutual information	–	–	–	Yes
[1]	Free-form image registration with PLSR deformation model	Cross correlation	8 patients (24 seq., 50 fr.), 5 healthy (15 seq., 50 fr.), synthetic data	SI-curves	Yes	Yes
[35]	PLSR Deformation model	–	9 patients (27 seq., 50 fr.), synthetic data	Motion reduction	Yes	–
[36]	Translation based registration	Cross correlation	10 patients (4–6 slices, 433 frames in total)	Motion reduction, SI-curves	No/yes <sup>a</sup>	5 sec pr. 50 fr.
[14]	Translation based registration	Mutual Information	5 patients (15 seq., 30 fr.)	Histograms, SI-curves, perf. maps, motion reduction	Yes	8 min pr. 30 fr.
[61]	Sensitivity analysis on perf. assessment	–	9 seq.	Comparison of perfusion parameters	–	–

<sup>a</sup>Breath-holding while comfortable, free-breathing after that.

Gupta et al. [36], Bansal and Funka-Lea [5] and Bracoud et al. [14] use a translation based registration method. Gupta et al. with cross-correlation and the latter two with mutual information (MI) as a similarity measure. None of the methods compensates for rotation or deformable shape changes. Gupta et al. validate on 10 patients, measuring motion reduction and comparing SI-curves. The computation time of 50 frames sequence is 5 seconds but 30 out of 433 frames needed manual post processing. Bracoud et al. evaluate their method on 5 patients (3 slices, 30 frames per sequence) with stable CAD. They compare histograms before and after registration, inspect SI-curves and perfusion maps and measure motion reduction. Their method takes 8 minutes for a 30 frame sequence including a manual definition of a rectangular ROI in each image of the sequence.

Spreeuwers et al. [61] present a refinement method to optimally place the myocardial borders after manually (or automatically) tracing them. They investigate the effect of displacements of the myocardial on perfusion analysis and use that information to correct the boundaries. They apply their method on 9 image sequences on which the myocardial has been traced, and compare perfusion parameters before and after correction. This method could be applied to improve results obtained from any of the approaches discussed above.

Table 3.1 gives an overview of the registration methods stated above.

## 3.2 Summary and next steps

This chapter introduced different approaches for automatic registration of myocardial perfusion data. The registration method used in this study is based on statistical modelling of shape and texture of the images. The difference between this method and the methods assessed in this chapter will be discussed in the end of this thesis, in Chapter 10. Next chapter will provide the necessary definitions and basic concepts for statistical shape and texture modelling.

## Chapter 4

# Preliminaries for statistical appearance modelling

This chapter provides a basis for statistical shape and texture modelling. Firstly, definitions of shape, texture, landmarks and shape representation are given. Secondly, principal components analysis (PCA) for shape data is addressed and thirdly, alignment of 2D shapes by planar Procrustes analysis is discussed. For the reader familiar with statistical modelling of shape and texture this chapter can be skipped.

### 4.1 Definitions of shape and texture

The *appearance* of an object is a combination of its shape and texture. This section provides definitions of those.

One of the most common definitions of a shape is due to D.G. Kendall (1977) [27]:

*Shape* is all the geometrical information that remains when location, scale and rotational effects are filtered out from an object.

This means that two objects are of same class of shapes if they can be rotated, translated and scaled so that they exactly match each other.

In the framework of statistical models of appearance, texture is defined as follows [20]:

*Texture* is the pattern of intensities or colours across an object.

## 4.2 Landmarks

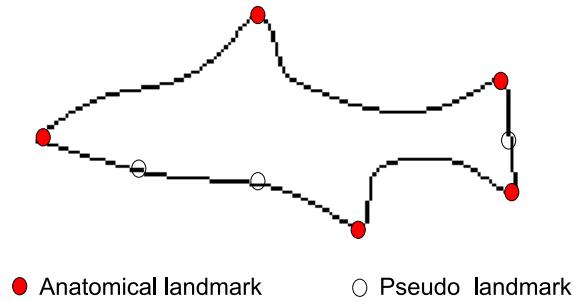
A shape can be defined by a finite number of points called *landmarks*. For a set of shapes belonging to the same class or population, a landmark is defined as follows [27]:

A *landmark* is a point of correspondence on each object that matches between and within populations.

Landmarks can be classified into three types:

- *Anatomical landmark*: A point assigned by an expert that corresponds between objects in a biologically meaningful way.
- *Mathematical landmark*: A point located on an object according to some mathematical or geometrical property, e.g. at a point of high curvature.
- *Pseudo landmark*: A point located either around the outline of an object or in between anatomical or mathematical landmarks.

An example of anatomical and pseudo landmarks is given in Figure 4.1.



**Figure 4.1:** Anatomical landmarks (which in this case also apply as mathematical landmarks) and pseudo landmarks placed equidistantly between the anatomical, illustrated on a fish shape.

A shape  $\mathbf{s}$  can be represented by the coordinates of its landmarks. Two types of representations of two-dimensional shapes are applied in this project. One is to arrange them in a column vector of size  $2n_l$  where  $n_l$  is the number of landmarks.

$$\mathbf{s} = [ x_1 \ y_1 \ x_2 \ y_2 \ \dots \ x_{n_l} \ y_{n_l} ]^T \quad (4.1)$$

An alternative is to use complex representation

$$\mathbf{s} = [ x_1 + iy_1 \ x_2 + iy_2 \ \dots \ x_{n_l} + iy_{n_l} ]^T, \quad i = \sqrt{-1} \quad (4.2)$$

### 4.3 Principal component analysis

A *Principal component analysis (PCA)* was first introduced by Harold Hotelling in 1930 based on work by Karl Pearson [17]. PCA is used to reduce dimensionality of multivariate data. This is achieved by rotating the data set so that the variance is maximised.

Continuing with the shape framework let the data consist of  $n_s$  shapes represented by column vectors of length  $2n_l$  as in Equation 4.1 with mean

$$\bar{\mathbf{s}} = \frac{1}{n_s} \sum_{k=1}^{n_s} \mathbf{s}_k. \quad (4.3)$$

The variance maximisation is done by an eigenanalysis of the data's dispersion matrix (covariance matrix)  $\Sigma_s$  estimated by

$$\hat{\Sigma}_s = \frac{1}{n_s - 1} \sum_{k=1}^{n_s} (\mathbf{s}_k - \bar{\mathbf{s}})(\mathbf{s}_k - \bar{\mathbf{s}})^T. \quad (4.4)$$

$\hat{\Sigma}_s$  is symmetric and it is possible to determine a set of eigenvectors,  $\{\phi_k\}_{k=1}^{2n_l}$  which form an orthonormal basis,

$$\Phi_s = \begin{bmatrix} & & & \\ \phi_1 & \phi_2 & \dots & \phi_{2n_l} \\ & & & \end{bmatrix}. \quad (4.5)$$

Now it holds that

$$\hat{\Sigma}_s \Phi_s = \Lambda \Phi_s, \quad (4.6)$$

where  $\Lambda$  holds the eigenvalues ordered in decreasing order  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{2n_l}$ ,

$$\Lambda = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_{2n_l} \end{bmatrix}. \quad (4.7)$$

The eigenvectors define the principal components or principal axes on which the data is projected. All axes are orthogonal. The first principal axis  $\phi_1$  is defined so that it maximises the variance of the projected data. The second axis  $\phi_2$  is defined to maximise the remaining variance along that axis and so forth.

After performing PCA on a set of shapes, any of the shapes can be represented by

$$\mathbf{s} = \bar{\mathbf{s}} + \Phi_s \mathbf{b}_s, \quad (4.8)$$

where  $\mathbf{b}_s$  holds the shape model parameters which can be varied to generate different shapes.

The  $k$ 'th model parameter  $b_k$  has variance  $\lambda_k$  and typically instances similar to the ones modelled are assured by applying limits of  $\pm 3\sqrt{\lambda_k}$ , i.e. 3 standard deviations. For a univariate normal distribution this accounts for 99.73% of probability mass.

The total model variance is defined by the sum of the eigenvalues,

$$V_T = \sum_k \lambda_k. \quad (4.9)$$

Usually, the number of eigenvectors and corresponding eigenvalues to apply in the model is reduced to contribute to a given proportion of the total variance. This means that the  $n_m$  largest eigenvalues can be chosen so that

$$\sum_{k=1}^{n_m} \lambda_k = f_v V_T, \quad (4.10)$$

where  $f_v$  denotes the desired proportion of the total variance for the model to explain and  $n_m$  is also referred to as the number of *modes*.  $b_k$  is said to control the  $k$ 'th mode of variation.

## 4.4 Procrustes alignment

Referring to the definition of a shape, an essential step prior to shape PCA is to align the set of objects so as to filter out location, scale and rotational effects. This is typically done by Procrustes alignment.

Planar Procrustes alignment is discussed in [27]. In this 2D case, the complex notation given in Equation 4.2 is adopted. Two scenarios will be addressed here, the alignment of two shapes or *full ordinary Procrustes analysis* and the alignment of a set of shapes or *full general Procrustes analysis*. The term *full* here refers to that the shapes are aligned using the similarity transforms, i.e. translation, rotation and scaling.

#### 4.4.1 Planar ordinary Procrustes analysis

Consider two shapes  $\mathbf{s}_1$  and  $\mathbf{s}_2$  with  $n_l$  elements (landmarks) represented on the complex form. The shapes are assumed to be **centred** so that  $\mathbf{s}_k^* \mathbf{1}_{n_l} = 0$ ,  $k = 1, 2$ . Here  $\mathbf{s}_k^*$  denotes the transpose of the complex conjugate of  $\mathbf{s}_k$ .

The goal is to align the two shapes by similarity transformations. This problem can be defined by a regression equation,

$$\begin{aligned} \mathbf{s}_1 &= (a + ib)\mathbf{1}_{n_l} + \beta e^{i\theta} \mathbf{s}_2 + \boldsymbol{\epsilon}, \\ a + ib &\in \mathbb{C}, \quad \beta \in \mathbb{R}_+, \quad \theta \in [0, 2\pi[, \quad \boldsymbol{\epsilon} \in \mathbb{C}^{n_l}. \end{aligned} \quad (4.11)$$

The transformation parameters are  $a$  and  $b$  which account for the translation, the scaling parameter  $\beta$  and the angle of rotation  $\theta$ .  $\boldsymbol{\epsilon}$  denotes the error vector. The optimal fit is obtained by a minimisation of the sum of square errors,

$$\begin{aligned} D^2 &= \boldsymbol{\epsilon}^* \boldsymbol{\epsilon} \\ &= \|\mathbf{s}_1 - \mathbf{s}_2 \beta e^{i\theta} - (a + ib)\mathbf{1}_{n_l}\|^2. \end{aligned} \quad (4.12)$$

It can be derived [27] that the parameters are estimated as

$$\hat{a} + i\hat{b} = 0 \quad (4.13)$$

$$\hat{\theta} = \arg(\mathbf{s}_2^* \mathbf{s}_1) \quad (4.14)$$

$$\hat{\beta} = \frac{\sqrt{\mathbf{s}_2^* \mathbf{s}_1 \mathbf{s}_1^* \mathbf{s}_2}}{\mathbf{s}_2^* \mathbf{s}_2}. \quad (4.15)$$

By inserting this into Equation 4.11, we arrive at the *full Procrustes fit*,  $\mathbf{s}_2^P$ , of  $\mathbf{s}_2$  onto  $\mathbf{s}_1$ ,

$$\mathbf{s}_2^P = \frac{\mathbf{s}_2^* \mathbf{s}_1 \mathbf{s}_2}{\mathbf{s}_2^* \mathbf{s}_2}. \quad (4.16)$$

#### 4.4.2 Planar generalised Procrustes analysis

In the general case, the aim is to align a set of shapes  $\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_{n_s}$ . This is done by aligning each shape to the *full Procrustes mean shape*,  $\boldsymbol{\mu}$ , that must be estimated simultaneously. It can be shown that an estimate,  $\hat{\boldsymbol{\mu}}$ , of the Procrustes mean shape is given by the eigenvector corresponding to the largest eigenvalue of the matrix

$$\mathbf{S} = \sum_{k=1}^{n_s} \mathbf{z}_k \mathbf{z}_k^*, \quad (4.17)$$

where  $\mathbf{z}_k$  is the standardised shape,  $\mathbf{z}_k = \frac{\mathbf{s}_k}{\|\mathbf{s}_k\|}$ .

The full Procrustes fit of  $\mathbf{s}_k$  ( $k = 1, \dots, n_s$ ) onto  $\hat{\boldsymbol{\mu}}$  can be derived similarly to Equation 4.16,

$$\mathbf{s}_k^P = \frac{\mathbf{s}_k^* \hat{\boldsymbol{\mu}} \mathbf{s}_k}{\mathbf{s}_k^* \mathbf{s}_k}, \quad k = 1, \dots, n_s. \quad (4.18)$$

After a full Procrustes alignment of all shapes, the mean shape  $\hat{\boldsymbol{\mu}}$  is equivalent to  $\bar{\mathbf{s}}$  from Equation 4.3.

## 4.5 Summary and next steps

This concludes the preliminary discussion of statistical modelling of shape and texture. The terms shape and texture have been defined and landmarks and representation of shapes have been addressed. Additionally, statistical shape modelling by PCA and the alignment of shapes by Procrustes analysis have been discussed. The next three chapters are based on this discussion, where the first two deal with semi-automatic placing of landmarks and the third with an AAM based registration method.

## Chapter 5

# Tool for manual extraction of shape contours from images

One step towards a robust shape model is to generate a training set of shapes with good point correspondences. A semi-automatic training set formation is performed in two steps. First, object outlines are extracted from the images and secondly, landmark correspondences are optimised (by MDL shape modelling in our case). The first step, i.e. manual extraction of shape contours is the topic of this chapter. It deals with the implementation of a contour extraction tool intended to ease the manual work. The tool is especially constructed for the cardiac data but can easily be generalised for any kind of data.

The section is structured as follows: Firstly, the manual placing of marks to define the shape contours is considered, involving general concepts of the contour extraction tool. Secondly, theoretical discussion of cubic splines, used for the interpolation of the marks, is provided and thirdly, example images from the annotation tool are given. Finally, some conclusions are drawn. More information on the annotation tool (given as a help-file) is provided in Appendix B.1.

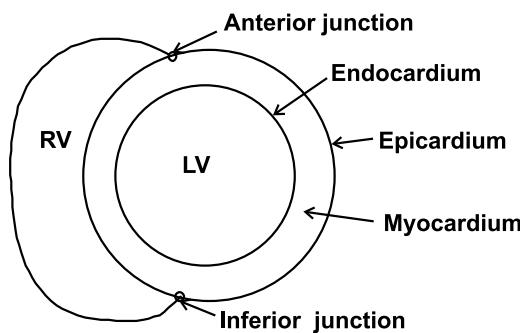
### 5.1 Contour extraction tool and placing of marks

The implemented contour extraction tool enables the user to extract the desired object by marking points on the contour by a mouse or a digitiser. Subsequently, the dense contour is given by an interpolating cubic spline between the points as discussed in section 5.2.

The program displays the image frames for one patient and one slice at a time by increasing frame number. For each image frame, there are three contours to annotate, i. e. the *epicardial*, *endocardial* and the *right ventricle(RV)* (see Figure 5.1). The three contours extracted from one image frame of one slice will be referred

to as the *cardiac shape* from this point forward. The cardiac shape includes two anatomical landmarks, the points where the RV border meets the epicardium, called *inferior junction* and *anterior junction* (see Figure 5.1). In order to simplify later steps in the shape modelling procedure, the anatomical landmarks are included in the annotation. With this in mind, the user is requested to annotate according to the following:

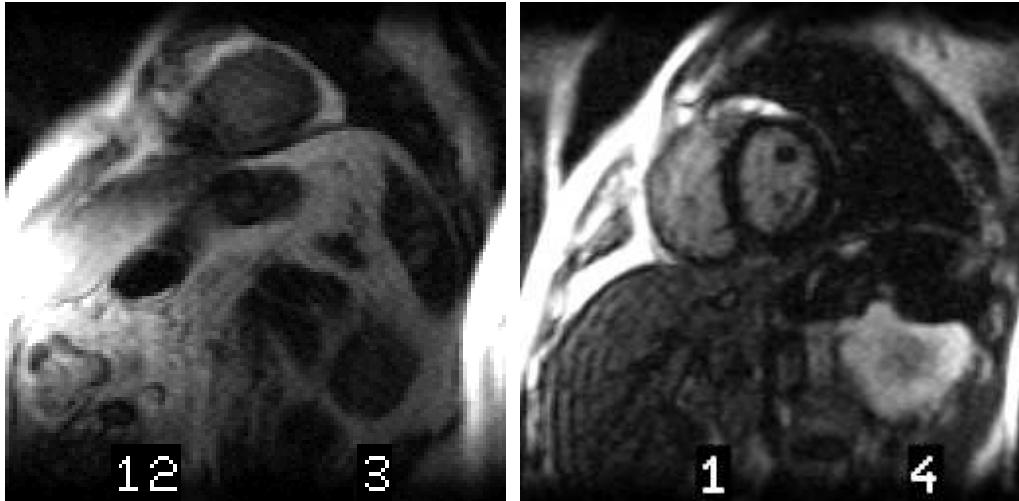
- Annotate epicardium, start at anterior junction
- Annotate endocardium, no restrictions on starting point
- Annotate RV border, start at anterior junction, finish at inferior junction



**Figure 5.1:** The cardiac shape with anatomical landmarks labelled: Anterior junction and inferior junction between right and left ventricle. The three contours annotated are endocardium, epicardium and the RV border.

Some of the cardiac images are of poor quality, especially those that occur before contrast injection (see comparison of poor quality versus average quality image in Figure 5.2). In this case, it can be fairly difficult to determine the myocardial borders. This problem can never be entirely overcome but to ease the detection of borders, the "resolution" of the images is increased by bilinear interpolation. Further, a bounding box is defined after annotating the first frame in a sequence and intensity-stretching is applied. The program then allows the user to fine-tune the appearance of the image while annotating by adjusting contrast and brightness. Moreover, the annotation of the latest frame is displayed as an assistance in determination of the borders.

Additional and essential features of the tool are that the user is able to undo last point or undo all points on the current contour. After annotating a shape, the user can determine whether to keep the annotation or repeat it.



**Figure 5.2:** Left: Image of poor quality where determination of myocardial boundaries is problematic. Right: Image of quality above average.

## 5.2 Interpolation of marks by cubic splines

After placing points on one contour (epicardial, endocardial or RV), it is parameterised assuming equidistant marks. Given the two sets of data points,  $\{t_k, x_k\}_{k=1}^n$  and  $\{t_k, y_k\}_{k=1}^n$  where  $t$  is the curve parameter and  $n$  is the number of data points, we want to define a function  $s(t)$ ,  $t \in [t_1, t_n]$  for each of the curves. To simplify, the following discussion applies to the first mentioned set of data points ( $x$ -coordinate).

The function should interpolate through the data points,

$$s(t_k) = x_k, \quad k = 1, \dots, n. \quad (5.1)$$

This could be done by polynomial interpolation or by linear interpolation. The first choice often generates "overswings" in the data and is not stable in the end points. The latter choice results in non-desirable kinks at the data points.

Instead, a smooth function, which has neither of the above mentioned disadvantages is desirable. This is satisfied by a natural cubic spline. A cubic spline is constructed of piecewise third-order polynomials across the set of knots, which here are set equal to the abscissae ( $t$ ). To obtain the smoothness property, the piecewise polynomials and its first two derivatives vary continuously across the set of knots,

$$s^{(p)}(t_k-) = s^{(p)}(t_k+), \quad p = 0, 1, 2 \quad k = 1, \dots, n. \quad (5.2)$$

Furthermore, for a natural cubic spline the second derivatives in the endpoints

should be equal to zero, reducing the risk of "overswings",

$$s''(t_1) = s''(t_n) = 0. \quad (5.3)$$

By using a moment representation of the spline<sup>1</sup> and demanding Equations 5.1–5.3 to hold, a linear system of equations can be derived.

$$\begin{bmatrix} 2(h_2 + h_3) & h_3 & & \\ h_3 & 2(h_3 + h_4) & h_4 & \\ & \ddots & \ddots & \\ & & h_{n-1} & 2(h_{n-1} + h_n) \end{bmatrix} \begin{bmatrix} s''_2 \\ s''_3 \\ \vdots \\ s''_{n-1} \end{bmatrix} = \begin{bmatrix} d_2 \\ d_3 \\ \vdots \\ d_{n-1} \end{bmatrix} \quad (5.4)$$

where

$$\begin{aligned} h_k &= t_k - t_{k-1}, \quad k = 2, \dots, n \\ d_k &= 6 \left( \frac{x_{k+1} - x_k}{h_{k+1}} - \frac{x_k - x_{k-1}}{h_k} \right), \quad k = 2, \dots, n-1 \\ s''_k &= s''(t_k). \end{aligned} \quad (5.5)$$

For a definition and further discussion of the moment representation of splines, refer to [51]. Writing the linear system in short gives

$$\mathbf{A}\mathbf{s}_{2:n-1}'' = \mathbf{d}_{2:n-1} \quad (5.6)$$

For the two closed contours (epicardium and endocardium), a periodic property is added to the spline,

$$s^{(p)}(t_n) = s^{(p)}(t_1), \quad p = 0, 1, 2. \quad (5.7)$$

---

<sup>1</sup>Alternative representations are B-spline representation and Hermite representation [51].

Adding this to the linear system gives

$$\mathbf{A}^{\text{periodic}} \mathbf{s}_{2:n}'' = \mathbf{d}_{2:n}, \quad (5.8)$$

where

$$\mathbf{A}^{\text{periodic}} = \begin{bmatrix} 2(h_2 + h_3) & h_3 & & h_2 \\ h_3 & 2(h_3 + h_4) & h_4 & \\ & \ddots & \ddots & \\ & & h_{n-1} & 2(h_{n-1} + h_n) & h_n \\ h_2 & & h_n & 2(h_n + h_2) & \end{bmatrix} \quad (5.9)$$

and

$$d_n = 6 \left( \frac{x_1 - x_n}{h_1} - \frac{x_n - x_{n-1}}{h_n} \right). \quad (5.10)$$

The built-in MATLAB function `csape` is used to calculate the spline. The function utilises MATLAB's '\' operator for solving the linear systems given in Equations 5.6 and 5.8. The operator solves a linear system  $\mathbf{Ax} = \mathbf{b}$  after checking the properties of the  $\mathbf{A}$  matrix to determine the appropriate solution algorithm.

For the system given in Equation 5.6,  $\mathbf{A}$  is tridiagonal, symmetric and positive definite (since it is obviously diagonal dominant). Consequently, the equation system is solved by simple Gaussian elimination (without pivoting).

The equation system for a periodic spline given in Equation 5.8 can be solved via *LU* factorisation as suggested in [51] or in a more general way using Cholesky factorisation as MATLAB's '\' operator.

The `csape` function allows representing the data points on the complex form

$$x_k + iy_k, \quad k = 1, \dots, n, \quad i = \sqrt{-1}. \quad (5.11)$$

This gives simultaneous interpolation of the two sets of data points and results in a complex representation of the spline

$$s(t) = s_x(t) + is_y(t). \quad (5.12)$$

An alternative to interpolation of the data points is to use curve fitting. This means that the data points are assumed to satisfy

$$x_k = f(t_k) + \epsilon_k \quad k = 1, \dots, n \quad (5.13)$$

where  $f$  is the socalled underlying function and  $\epsilon$  are the measurement errors. A curve fitting aims at describing the data by a model  $M(\mathbf{a}, t)$ , so that the residuals are minimised with respect to the parameters,  $\mathbf{a}$ , of the model,

$$\mathbf{a}^* = \operatorname{argmin}_{\mathbf{a}} \left\{ \sum_{k=1}^n x_k - M(\mathbf{a}, t) \right\}. \quad (5.14)$$

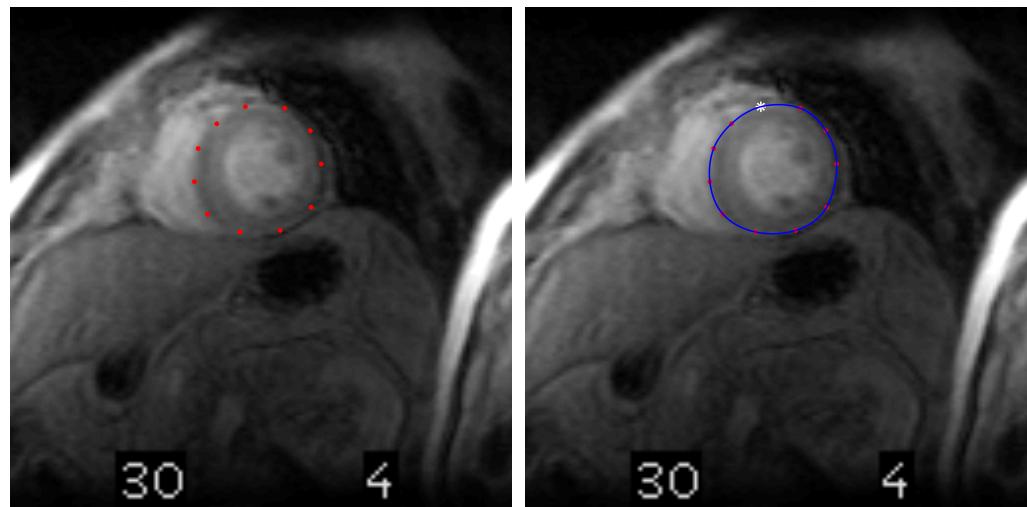
As a model ( $M$ ), one could e.g. use a fitting spline.

In this study, the data points were interpolated rather than fitted. Reasoning for this choice will be given in Section 5.4.

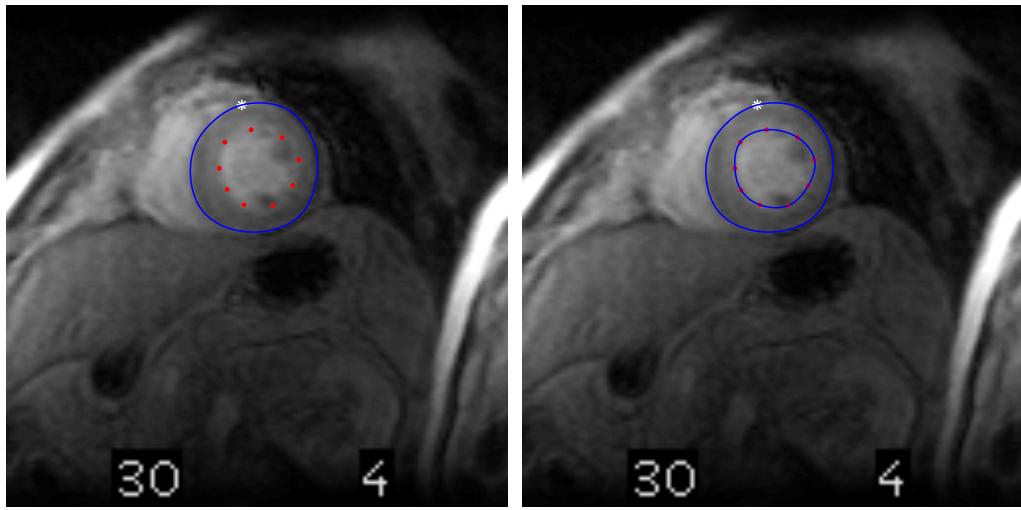
In order to save calculations in the MDL algorithm, the number of evaluation points for the splines was chosen carefully to represent the contour in approximately every second pixel.

### 5.3 Examples from the contour extraction tool

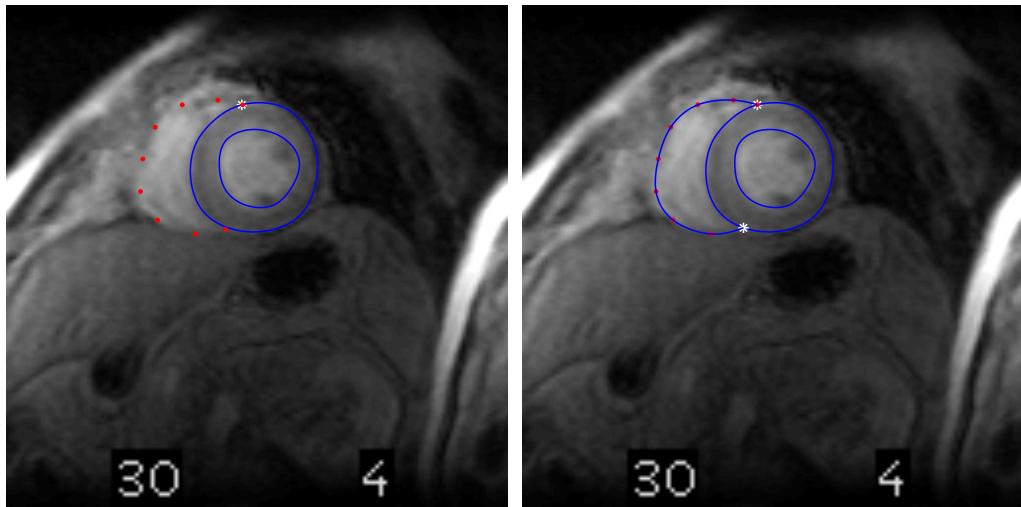
The contour extraction procedure is illustrated in Figures 5.3–5.5.



**Figure 5.3:** Annotation of epicardial. Left: Manual annotation. Right: Spline has been interpolated to the points. Anterior junction denoted by a white mark.



**Figure 5.4:** Annotation of endocardial. Left: Manual annotation. Right: Spline has been interpolated to the points. Anterior junction denoted by a white mark.



**Figure 5.5:** Annotation of RV. Left: Manual annotation. Right: Spline has been interpolated to the points. Inferior and anterior junction denoted by a white mark

## 5.4 Conclusions

The contour extraction of 2000 shapes is a quite time consuming procedure, more precisely, it took one week of normal working hours. However, getting rid of the point correspondence requirement as when generating a full training set simplifies matters considerably. Furthermore, using a digitiser (WACOM tablet) resulted in a faster and more accurate annotation than when using a mouse as examined (qualitatively) in preliminary testing of the program.

As expected, the more images that were annotated, the faster the annotation became (an example of a *learning curve*). The annotations were carried out patient by patient which could lead to biased results with respect to patients. This could have been compensated for by randomising the appearance of image frames, but to ease implementation and data storage, the annotations were performed as mentioned above.

Even though the displaying of the previous annotation is only intended to give a rough idea where to place the marks, especially for the bad quality images, the annotations can become too dependent on the previous ones and possibly result in a systematic error. Nevertheless this feature was considered an advantage of the program rather than a disadvantage, but it is important to be aware of those factors while annotating. Since the bad quality images mostly occur in early time frames, this feature could have been better utilised by displaying the frames in reverse order.

By using interpolation rather than curve fitting when generating a full contour from the marked points, the manual annotation errors (measurement errors) are accepted since the resulting contour goes through all manually placed points. By using a fitting spline, a balancing of measurement errors and approximation errors ( $\sum_{i=1}^n f(t_i) - M(\mathbf{c}, t)$ ) is obtained but possibly it misses some details of the annotation by considering them as errors.

Since a MATLAB code for a periodic fitting spline was not available and would have been a study in itself to implement properly, the built-in MATLAB implementation for an interpolating spline was preferred. However, the main point is that this decision is not considered crucial for the final goal of constructing a good appearance model.

## 5.5 Summary and next steps

This chapter has described the implementation of a tool to ease manual extraction of the cardiac shape from each frame of a multi-slice perfusion sequence. The tool allows the user to mark arbitrary points on the outline of the shape and subsequently provides the shape contours by an interpolating cubic spline. Provided the cardiac shape outlines, landmarks can automatically be placed by MDL shape modelling. This is the subject of next chapter.

## Chapter 6

# Minimum description length shape modelling

The placing of landmarks on a training set of shapes is crucial for every shape model. This step is often a manual procedure which can be very tedious, give erroneous results and a considerable variation dependent on the person placing the landmarks. This section deals with the automatic placing of landmarks given the manually extracted outline of the shape.

Firstly, a brief review of different approaches made to automate the landmarking procedure is given. Secondly, the general theory of MDL shape modelling and derivation of the description length of a shape model is addressed and finally, the adjustments needed for the cardiac data to adapt to the MDL framework are covered.

### 6.1 Review of automatic model building approaches

Several attempts have been made to make the landmarking procedure automatic or semi-automatic.

A fully automatic placement of landmarks addresses the generation of a training set directly from the set of images. Most attempts consider registration of the set of images to a previously annotated reference image, optimising a similarity measure between images [55], [31], [33].

A semi-automatic placing of landmarks means that given the desired object outlines from the images, landmarks are automatically placed.

Benayoun et al. [8], Kambhamettu and Goldgof [41] and Tagare [66] base their approaches on shape features, where high-curvature points are matched. They all provide pairwise correspondences and may therefore not lead to a global optimum.

Many approaches consider establishment of point correspondences as an optimisation problem [7, 10, 37, 54, 56, 24].

Rangarajan et al. [54] present the robust point matching algorithm, which extends the alignment of a set of shapes to solve for correspondences by model variance minimisation.

Bookstein [10] combines Procrustes analysis with thin-plate splines, minimising a bending-energy term.

Hill and Taylor [37] apply a minimisation of the total model variance combined with dissimilarities of shape features. This objective function results in many local minima.

Kotcheff and Taylor [42] obtain, to some extent, compact shape models by minimising the determinant of the model covariance but a theoretical justification for optimality is not given.

Sebastian et al. [56] use intrinsic properties of two curves to an energy minimisation framework. They introduce an alignment curve allowing a symmetric treatment of the two curves.

Belongie et al. [7] maximise similarity of so-called shape contexts to solve the point correspondence problem between two shapes. The shape context descriptor captures global properties of the shape relative to a single reference point.

The two last mentioned approaches suffer from the fact that they give pairwise correspondences rather than a global solution.

The MDL approach first introduced by Davies et al. in 2001 [24] obtains point correspondence for a set of shapes by minimising the description length, a term adopted from information theory. The description length in this case is the cost of transmitting the PCA coded model parameters in addition to the transmission cost of the encoded data values. This means that MDL balances the complexity of the model against how well the model fits the data. This approach will be used for an automatic model building of the cardiac shape contours. Next sections will explain the concepts of MDL shape modelling.

## 6.2 Basic concepts of MDL shape modelling

Now, we arrange the problem of obtaining point correspondences to a mathematical framework. Firstly, properties of a parameterisation function defined for each shape are addressed. Secondly, a derivation of the objective function is given in Section 6.2.2. Techniques for minimisation of the objective function are addressed in Section 6.2.3 with special focus on the implementation by Thodberg [68]. Finally, a case study will be given in order to illustrate how powerful the method can be.

### 6.2.1 Parameterisation function

The set of shapes can be represented by parametric curves:

$$\mathbf{s}_k(t) = (\mathbf{s}_{xk}(t), \mathbf{s}_{yk}(t)), t \in [0, 1], k = 1, \dots, n_s \quad (6.1)$$

where  $n_s$  is the number of shapes. For each training shape  $\mathbf{s}_k, k = 1, \dots, n_s$ , a parameterisation function  $\psi_k$  of the shape boundaries is defined. The aim is to choose  $\psi_k$  so that the best possible statistical shape model is obtained from the training set.

To preserve the initial point ordering, the parameterisation  $\psi_k(t)$  must be a monotonically increasing function of  $t$ . That is:

$$s_k(t) \rightarrow s_k(t'), t' = \psi_k(t), \psi_k : [0, 1] \rightarrow [0, 1] \quad (6.2)$$

where  $\psi_k$  must be one-to-one, onto and invertible, i.e. a diffeomorphic mapping.

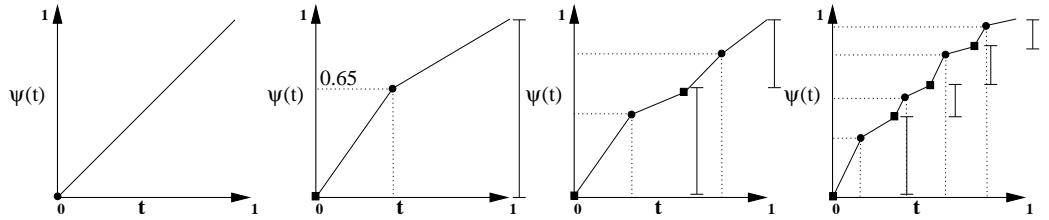
Davies et al. [24] suggest a piecewise linear parameterisation of each shape. It is defined by recursively inserting nodes between those already present. This hierarchical representation constrains the positions of the child nodes to lie in the range  $[0, 1]$  of fractional distances where the values 0 and 1 indicate that the child node is positioned on its left or right neighbour respectively. In this representation an equidistant parameterisation of a shape would place every child node in the center of its two parent nodes, i.e. every child node would have the value 0.5 with respect to its parents and the parameterisation function would be a straight line. The hierarchical property infers that the parameterisation must be based on  $2^Q + 1$  nodes, where  $Q \in \mathbb{Z}^+$  is the number of levels of the hierarchy.

A parameterisation  $\psi$ , based on  $Q = 3$  levels may be implicitly denoted as

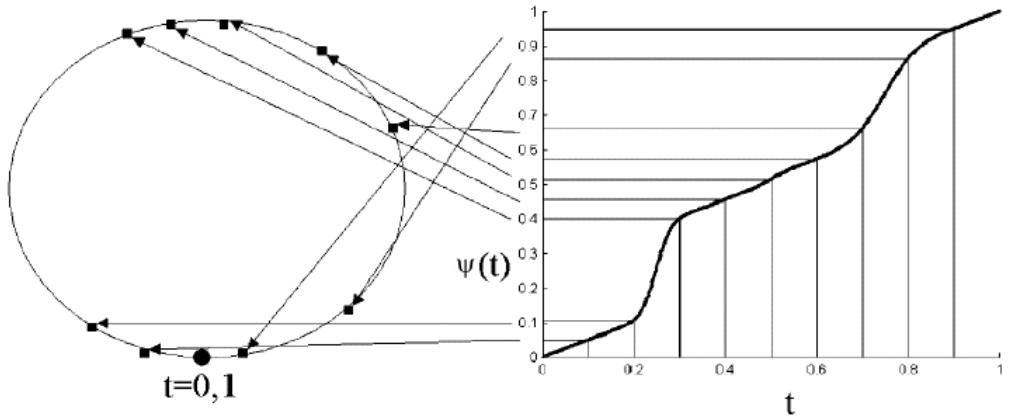
$$\psi(\tau_{11}(\tau_{21}(\tau_{31}, \tau_{32}), \tau_{22}(\tau_{33}, \tau_{34}))) \quad (6.3)$$

where  $\tau_{ij}$  is the fractional distance of the  $j$ -th child node on level  $i$  to its parents from level  $i - 1$ .

The recursive formation of the parameterisation function is illustrated in Figure 6.1. After the formation of a parameterisation function for a shape, an arbitrary number of points can be sampled equidistantly from the function to represent the given shape. This is shown for a circle in Figure 6.2.



**Figure 6.1:** Recursive representation of reparameterisation. Squares denote parent nodes and circles denote child nodes. Distance between parents, i.e. the interval  $[0,1]$  to place the child node is also illustrated. The parameterisation for this example in terms of fractional distances is  $\psi(0.65(0.65(0.8, 0.4), 0.7(0.2, 0.5)))$ . By courtesy of Davies et al. [25].



**Figure 6.2:** An illustration of the sampling of a circle according to a given parameterisation function,  $\psi$ . By courtesy of Davies et al. [25].

### 6.2.2 Objective function

Now that we have a representation of the parameterisation functions, we turn back to the problem of selecting the parameterisation which results in the best statistical shape model. Consider the philosophy of Ockham's razor: *The simplest description is truer*. In relation to the quality of a statistical shape model, this means that the more *compact* the model is, the better it is. In this connection, *better* means the model's ability to *generalise* to unseen instances of the shape class, still preserving the model's *specificity*, i.e. the ability to stick to valid instances of the class.

A compactness of a model can be quantified by applying the Minimum Description Length (MDL) principle. This is a quantity from information theory and is based on the cost of transmitting data coded with respect to a set of parametric statistical models.

In the framework of statistical shape models, this means that the cost describes the

information needed to transmit the PCA representation of all the  $n_l$ -dimensional shapes in the training set, given  $n_m$  number of modes:

$$\mathbf{s}_k = \bar{\mathbf{s}} + \sum_{m=1}^{n_m} \boldsymbol{\phi}_m b_k^m. \quad (6.4)$$

The set of eigenvectors  $\Phi$  define an  $n_m$  dimensional *shape space* where the  $m$ -th coordinate is given by

$$b_k^m = (\mathbf{s}_k - \bar{\mathbf{s}})^\top \boldsymbol{\phi}_m. \quad (6.5)$$

The description length of a set of shapes is derived in Davies [23] and the remainder of this section addresses that derivation, except for the final objective function given in Equation 6.28 which is an approximation by Thodberg [68].

The orthonormal property of the eigenvectors in Equation 6.5 gives that the total description length  $\mathcal{L}_{total}$  can be calculated as

$$\mathcal{L}_{total} = \sum_{m=1}^{n_m} \mathcal{L}^m, \quad (6.6)$$

where  $\mathcal{L}^m$  is the description length of the 1D data set  $\mathcal{B}^m = \{b_k^m : k = 1 \dots n_s\}$ . This means that the description length can be calculated independently for each principle direction  $m$  as derived in the following.

### Calculation of the description length

Using Shannon's codeword length [57], a value  $\hat{\alpha}$  encoded with probability density function  $\mathcal{P}$  has the description length

$$\mathcal{L} = \log(\mathcal{P}(\hat{\alpha})). \quad (6.7)$$

The logarithm is 2-based to obtain the description length in bits. In order to utilise this definition, the data must be quantised to an accuracy  $\Delta$  by

$$b_k^m \rightarrow \hat{b}_k^m, \hat{b}_k^m = n\Delta, n \in \mathbb{Z}, \quad (6.8)$$

since a real number requires infinite amount of information to be described. Furthermore, the range of the data must be known. The original shapes' coordinates must have a strict upper-bound  $r$  such that

$$-\frac{r}{2} \leq s_{k\alpha} \leq \frac{r}{2}, \alpha = 1, \dots, n_l, k = 1, \dots, n_s. \quad (6.9)$$

This can be transformed to give an upper-bound,  $R$  of the data in shape space:

$$R = r\sqrt{n_l}, \quad b_k^m \leq R, \quad k = 1, \dots, n_s, \quad m = 1, \dots, n_m. \quad (6.10)$$

Using the fact that our data has zero mean in each principal direction  $m$ , it can be assumed to be encoded using a set of one-parameter Gaussian models,

$$f(b^m; \sigma^m) = \frac{1}{\sigma^m \sqrt{2\pi}} \exp\left(-\frac{(b^m)^2}{2(\sigma^m)^2}\right) \quad (6.11)$$

The description length can be divided into the description length of the encoding model and the description length of the data,

$$\mathcal{L} = \mathcal{L}_{parameters} + \mathcal{L}_{data}. \quad (6.12)$$

### Coding the parameters

The only encoding model parameter from the Gaussian model is  $\sigma$ , estimated from the quantised data values,

$$\sigma = \sqrt{\frac{1}{n_s} \sum_{i=1}^{n_s} \hat{b}_i^2}. \quad (6.13)$$

$\sigma$  is now quantised to some accuracy  $\delta$  to be as close as possible to the estimate in Equation 6.13:

$$\hat{\sigma} = n\delta, \quad n \in \mathbb{Z}^+. \quad (6.14)$$

It can be assumed that  $\hat{\sigma}$  is uniformly distributed over  $[\sigma_{min}, \sigma_{max}]$ . Inserting the uniform distribution to Equation 6.7 gives the description length of  $\hat{\sigma}$ ,

$$\mathcal{L}_{\hat{\sigma}} = \log\left(\frac{\sigma_{max} - \sigma_{min}}{\delta}\right). \quad (6.15)$$

Now, for the receiver to decode the message, the description length of  $\delta$  must be known. Given that  $\delta = 2^k$ ,  $k \in \mathbb{Z}$ , its description length can be calculated as

$$\mathcal{L}_{\hat{\delta}} = 1 + |\log \delta|. \quad (6.16)$$

Combining Equations 6.15 and 6.16 now gives

$$\begin{aligned}\mathcal{L}_{parameters} &= \mathcal{L}_{\hat{\sigma}} + \mathcal{L}_{\hat{\delta}} \\ &= \log\left(\frac{\sigma_{max} - \sigma_{min}}{\delta}\right) + 1 + |\log \delta| .\end{aligned}\quad (6.17)$$

### Coding the data

The probability of a value lying in the range  $\hat{b} \pm \frac{\Delta}{2}$ , using the Gaussian model in Equation 6.11 can be approximated by

$$\mathcal{P}(\hat{b}) = \int_{k=\hat{b}-\frac{\Delta}{2}}^{\hat{b}+\frac{\Delta}{2}} f(k; \hat{\sigma}) dk \approx \frac{\Delta}{\hat{\sigma}\sqrt{2\pi}} \exp\left(-\frac{\hat{b}^2}{2\hat{\sigma}^2}\right) . \quad (6.18)$$

By inserting this approximation to Equation 6.7, the description length of the data for one shape mode summed over all shapes is obtained:

$$\mathcal{L}_{data} = - \sum_{i=1}^{n_s} \log(\mathcal{P}(\hat{b}_i)) = -n_s \log \Delta + \frac{n_s}{2} \log(2\pi\hat{\sigma}^2) + \frac{1}{2\hat{\sigma}^2} \sum_{i=1}^{n_s} \hat{b}_i^2 \quad (6.19)$$

An inspection of the errors due to the approximation in Equation 6.18 leads to the choice of lower bound of  $\hat{\sigma}$ ,  $\sigma_{min} = 2\Delta$  (see [23]). With no prior knowledge,  $\hat{\sigma}$  is assumed to have the same range as the data, leading to the upper bound of  $\sigma_{max} = R$ .

Now, one must consider the true values of  $\sigma$  and  $b$ , (i.e. not the quantised values) in connection with Equation 6.19. Three cases will be addressed:

**Case 1:**  $\sigma > \sigma_{min}$ : Explicitly code the data

**Case 2:**  $\sigma \leq \sigma_{min}$ : Estimate  $\sigma$  with  $\sigma_{min}$

**Case 3:**  $B \leq \Delta$ : All data has the same quantised value.

**Case 1:** Assuming a uniform distribution, the expected values of  $\frac{1}{\hat{\sigma}^2}$  and  $\log(\hat{\sigma})$  can be approximated by

$$\begin{aligned}\mathbf{E}\left[\frac{1}{\hat{\sigma}^2}\right] &\approx \frac{1}{\sigma^2} \left(1 + \frac{\delta^2}{4\sigma^2}\right) \\ \mathbf{E}[\log(\hat{\sigma}^2)] &\approx \log(\sigma^2) - \frac{\delta^2}{12\sigma^2} .\end{aligned}\quad (6.20)$$

Substituting this and Equation 6.13 into Equation 6.19 gives

$$\mathcal{L}_{data}^{(1)} = -n_s \log \Delta + \frac{n_s}{2} \log(2\pi\sigma^2) + \frac{n_s}{2} + \frac{n_s \delta^2}{12\sigma^2} . \quad (6.21)$$

The optimal accuracy of  $\sigma, \delta^*$  can be found by setting the derivative of  $\mathcal{L}_{total}$  with respect to  $\delta$  to zero. This gives  $\delta^*(\sigma, n_s) = \min(1, \sigma\sqrt{\frac{12}{n_s}})$  and can be inserted directly into Equation 6.21.

**Case 2:** In this case  $\hat{\sigma}$  takes value below  $\sigma_{min}$  which has the effect that the approximation in Equation 6.18 becomes inaccurate. Nevertheless, it results in an overestimate of the description length and can therefore still be used.  $\sigma$  is estimated by  $\sigma_{min}$  and  $\delta^*$  is chosen by  $\min(1, \sigma_{min}\sqrt{\frac{12}{n_s}})$ . The substitution from Equation 6.13,  $\sum_{i=1}^{n_s} \hat{b}_i^2 = n_s \sigma^2$  is not valid in this case so we arrive at

$$\mathcal{L}_{data}^{(2)} = -n_s \log \Delta + \frac{n_s}{2} \log(2\pi\sigma_{min}^2) + \frac{1}{2\sigma_{min}^2} \sum_{i=1}^{n_s} \hat{b}_i^2. \quad (6.22)$$

**Case 3:** In this case, all the data,  $\mathcal{B}$  has the same quantised value, i.e.  $\mathcal{B} \leq \Delta$ . This means that only the position of the data, i.e. the mean needs to be sent. Since the mean is always zero, this costs nothing to describe, i.e.

$$\mathcal{L}_{data}^{(3)} = 0. \quad (6.23)$$

### The total description length

In order to simplify the final expression, it is noted that in the limit  $\Delta \rightarrow 0$ , the quantised values approach the original values,  $\hat{b} \rightarrow b$  and  $\hat{\sigma}^2 \rightarrow \frac{1}{n_s} \sum_{i=1}^{n_s} \hat{b}_i^2$ . Furthermore, given a sufficiently large number of shapes,  $n_s > 12\sigma^2$ , the optimal accuracy becomes  $\delta^* = \sigma\sqrt{\frac{12}{n_s}}$ .

Given this and adding the parameter description length in Equation 6.17 to each of the two non-zero cases for data description length we obtain for each direction  $m$

$$\begin{aligned} \mathcal{L}^{(1)}(\sigma_m, n_s, R, \Delta) &= \mathcal{L}_{data}^{(1)} + \mathcal{L}_{parameters} \\ &= \log(\sigma_{max} - \sigma_{min}) + 1 - 2\log\left(\sqrt{\frac{12}{n_s}}\right) \\ &\quad - n_s \log \Delta + \frac{n_s}{2} \log(2\pi) + \frac{n_s}{2} \\ &\quad - 2\log(\sigma_m) + n_s \log(\sigma_m) \end{aligned} \quad (6.24)$$

and

$$\begin{aligned}
\mathcal{L}^{(2)}(\sigma_m, n_s, R, \Delta) &= \mathcal{L}_{data}^{(2)} + \mathcal{L}_{parameters} \\
&= \log(\sigma_{max} - \sigma_{min}) + 1 - 2\log\left(\sqrt{\frac{12}{n_s}}\right) \\
&\quad - n_s \log \Delta + \frac{n_s}{2} \log(2\pi) - 2\log(\sigma_{min}) \\
&\quad + \frac{n_s}{2} \log(\sigma_{min}^2) + \frac{n_s}{2} \left(\frac{\sigma_m}{\sigma_{min}}\right)^2.
\end{aligned} \tag{6.25}$$

The total description length can now be derived by inserting the two equations above to Equation 6.6:

$$\begin{aligned}
\mathcal{L}_{total} &= F(n_s, R, \Delta) + \sum_{p=1}^{n_g} (n_s - 2) \log(\sigma_p) + \frac{n_s}{2} \\
&\quad \sum_{q=n_g+1}^{n_g+1+n_{min}} (n_s - 2) \log(\sigma_{min}) + \frac{n_s}{2} \left(\frac{\sigma_q}{\sigma_{min}}\right)^2,
\end{aligned} \tag{6.26}$$

where  $n_g$  is the number of directions for which the first case ( $\sigma_m > \sigma_{min}$ ) holds and  $n_{min}$  is the number of directions where the second case ( $\sigma_m \leq \sigma_{min}$ ) holds.  $F(n_s, R, \Delta)$  is constant for each training set

### Final objective function

When optimising the objective function, one should focus on the terms which change. This reduces Equation 6.26 to

$$\mathcal{L}_{total} = \frac{n_s}{2} \left( \sum_{p=1}^{n_g} (\log(\sigma_p^2) + 1) + \sum_{q=n_g+1}^{n_g+1+n_{min}} \log(\sigma_{min}^2) + \left(\frac{\sigma_q}{\sigma_{min}}\right)^2 \right), \tag{6.27}$$

where  $n_s - 2$  has been approximated by  $n_s$  assuming sufficiently large number of shapes.

By subtracting the constant  $(n_g + 1 + n_{min}) \frac{n_s}{2} \log(\sigma_{min}^2)$  from Equation 6.27, substitute  $\sigma_i^2$  by  $\lambda_i$  and the constant  $\sigma_{min}^2$  by  $\lambda_{cut}$ , we arrive at the final expression for the total description length as stated by Thodberg [68].

$$\mathcal{L}_{total} = \sum_{p=1}^{n_g} \left( \log\left(\frac{\lambda_p}{\lambda_{cut}}\right) + 1 \right) + \sum_{q=n_g+1}^{n_g+1+n_{min}} \left( \frac{\lambda_q}{\lambda_{cut}} \right) \tag{6.28}$$

Here,  $\frac{n_s}{2}$  is ignored since this is constant during optimisation. The main advantages of this objective function, compared to the previously stated ones, is that it has continuous derivatives and it is easier to understand.

Having quantified compactness by the description length, its minimum gives the most compact and thereby a model with good generalisation and specificity abilities as discussed before.

### 6.2.3 Minimisation of the description length

The task of minimising the objective function given in Equation 6.28 is non-trivial. Davies both tried genetic algorithms and a simplex-based local search algorithm where the latter gave better results. Ericsson and Åström [30] showed that steepest descent based optimisation gave more efficiency. In this study, the freely available code from Thodberg [68] based on pattern search optimisation is applied. For later discussion a description of his algorithm follows. In the following discussion, Description Length,  $DL$  is equivalent to  $\mathcal{L}_{total}$ .

The approach is to adjust the parameterisation function on one shape at a time according to an adaptive step length. To ease computations,  $\tau_{ij}$  (see Equation 6.3) is only adjusted for the marks on the first few levels. Those marks are referred to as *nodes* and the notation  $\tau(node)$  is adopted. The value of  $\tau$  for the remaining levels is kept constant (0.5).

After adjusting the parameterisation the corresponding shape marks are extracted, the shapes in the set are aligned by Procrustes alignment, PCA is performed and the description length is calculated. For each move, the description length is compared to the current best one. Based on the comparison, this move is either accepted or rejected. This is clarified in Table 6.1.

#### Additional remarks on Thodbergs algorithm

The stopping criteria of Thodbergs algorithm is only defined by the maximum number of passes. Here, a convergence measure was added, i.e. the number of times the objective value is allowed to be approximately the same as the value from last pass.

The optimisation includes a mechanism to prevent the marks to pile up at some regions and dilute at others. It is based on adding a stabilising term (see Equation 6.29) to the objective function. The term measures how far the average node parameter value across the set of shapes moves from some target node parameter, which typically represents equidistant parameterisation  $\tau_{node}^{target} = 0.5$ . The weight of the term is controlled by varying the value of the tolerance measure,  $T$ .

**Table 6.1:** Pseudo-code for Thodbergs MDL optimisation algorithm [68].

1:	Loop over passes
2:	Loop over nodes
3:	Loop over 5 steps
4:	Loop over shape examples
5:	Loop over $\pm$ stepsize
6:	Probe $\tau(\text{node}) = \tau(\text{node}) \pm \text{stepsize}$ of example
7:	Extract marks from parameterisation
8:	Center and normalise set
9:	Do Procrustes of set
10:	Do PCA of set
11:	Compute DL from PCA
12:	If $\text{DL} < \text{MDL}$ , set $\text{MDL} \leftarrow \text{DL}$ and break loop
13:	Undo $\tau(\text{node})$ change
14:	End of $\pm$ stepsize loop
15:	End of example loop
16:	Adjust stepsize
17:	End of step loop
18:	End of node loop
19:	End of passes loop

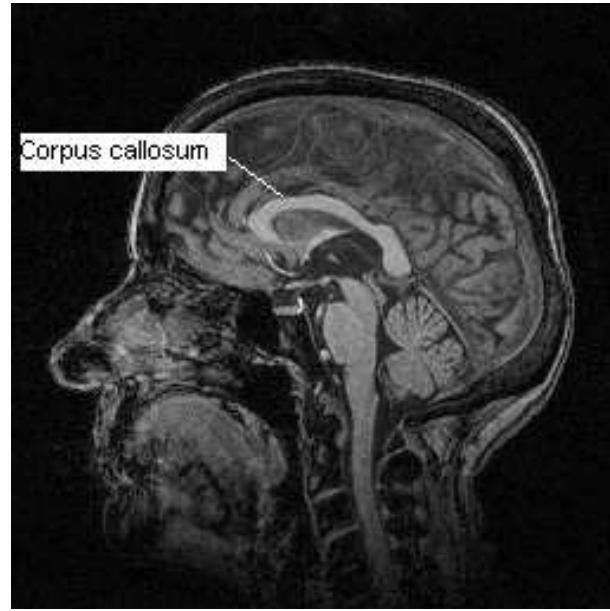
$$\sum_{\text{node}=1}^{\#\text{nodes}} \frac{\left( \tau_{\text{node}}^{\text{mean}} - \tau_{\text{node}}^{\text{target}} \right)^2}{T^2} \quad (6.29)$$

After doing the MDL optimisation, the result is fine-tuned by adding a curvature term to the cost function which measures deviations from mean curvature. This has given good results for examples with high-curvature landmark positions as head silhouettes [69].

#### 6.2.4 Corpus callosum case

In order to show how powerful the MDL approach can be, a case study is introduced. The *corpus callosum* is the area of the human brain which connects and enables communication between the two cerebral hemispheres. Many neurological studies indicate that the size and shape of the corpus callosum are related to gender, age, neurodegenerative diseases et cetera [10, 28, 46]. The gold standard for such studies is MRI, which allows acquisition of accurate images of the anatomy and function of the human brain [64]. An example image is given in Figure 6.3.

For the analysis of the MR brain images, a segmentation of the corpus callosum is crucial. In this relation, AAM has been applied with good results. However, the results of the AAM are heavily dependent on the landmarking. In this context, the MDL approach has given remarkable results [64]. To illustrate this, the first



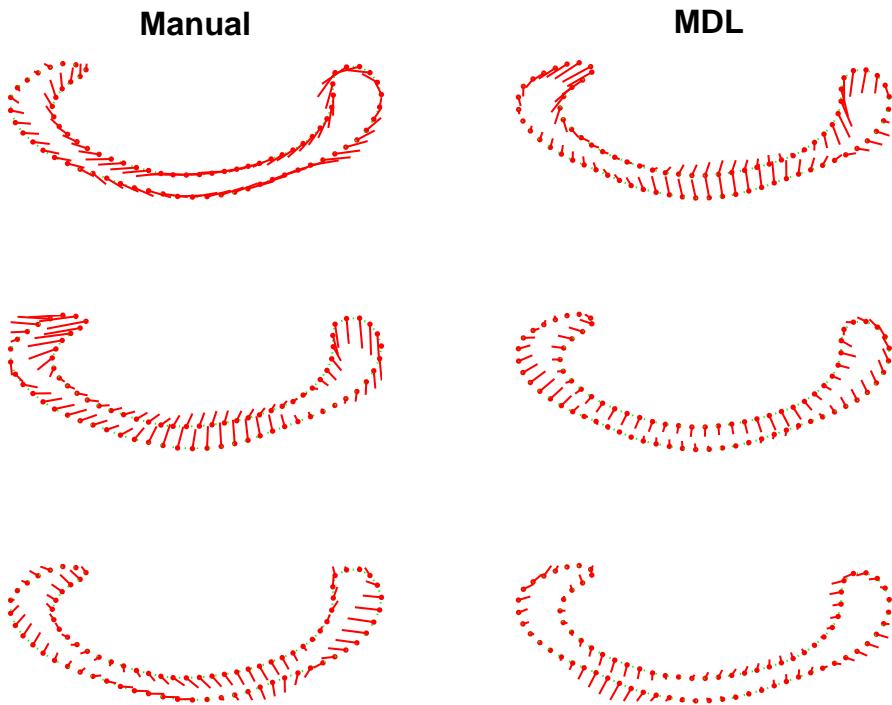
**Figure 6.3:** MR brain image with corpus callosum labelled.

three modes from PCA of shapes with manually placed landmarks is compared to shapes with the optimised MDL landmarks in Figure 6.4. The figure shows that the model is considerably better for the MDL optimised landmarks than for the manually placed landmarks. Compared to the manual model, the first mode for the MDL solution is much more specific on the shape variation itself, it lies orthogonal to the mean shape, which indicates a good shape model.

A quantitative measure of the compactness of the model is given by the total variance,  $V_T$  (refer to Equation 4.10). For the corpus callosum example, the total variance was calculated as:

$$\begin{aligned} \text{Manual model: } V_T &= 0.0087 \\ \text{MDL model: } V_T &= 0.0038 \end{aligned}$$

This shows that the total variance has been reduced by **56%** from the manual model to the MDL model. This extreme improvement suggests that it is a good idea to consider MDL in every shape model.



**Figure 6.4:** Figure 6.4: First three principal modes (mode 1–3 from top to bottom). The whiskers denote 3 standard deviations. Left: Manual model. Right: MDL model.

### 6.3 Adjustments of cardiac shape contours to the MDL framework

Having seen the promising results from the corpus callosum case, the MDL technique now seems an ideal way to place landmarks on the previously defined cardiac shapes. However, a few adjustments need to be made prior to applying the method. Before describing the actual adjustments, a discussion on data modelling is provided.

#### 6.3.1 Modelling issues

The shape modelling approach used here is to pool inter-patient and inter-slice variability of the set of shapes. This means that each cardiac shape is considered as one observation and by MDL optimisation, correspondence is obtained across slices and patients. This is different from what is done in the multi-slice AAM described in next chapter where each multi-slice frame is considered as one observation.

The reason for this inconsistency is that the multi-slice part of the AAM was added at the final stages of this project and modelling the MDL in this way was

considered adequate for a single-slice as well as the resulting multi-slice AAM.

Now the adjustments of the cardiac shape data to the MDL framework will be addressed.

### 6.3.2 Splitting up cardiac shape contours

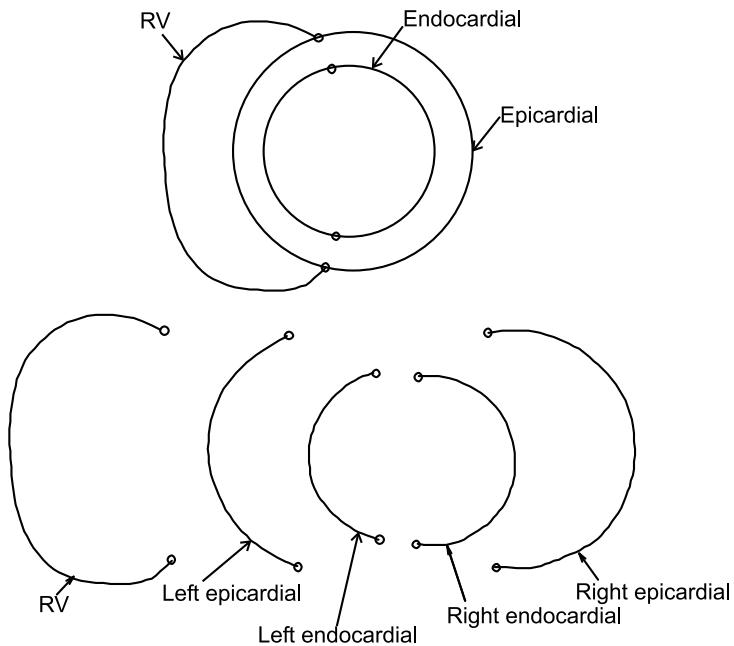
As mentioned earlier, the cardiac shape consists of three contours (curves), two closed (endocardial and epicardial) and one open (RV). The MDL method is not designed to deal with more than one set of shape contours, since it requires each shape contour in the set to be parameterised. Consequently the cardiac shape should be split up.

However, splitting the cardiac shape into the three above mentioned contours gives rise to some problems. This is due to the anatomical landmarks on the cardiac shape where the RV meets the epicardial (refer to Figure 5.1). That is, if the epicardial contour was to be optimised, those two landmarks would have to be kept fixed throughout the optimisation process. Keeping a point, which is neither a start nor an endpoint, fixed is problematic due to the hierarchical properties of the parameterisation function discussed in Section 6.2.1.

The approach used here will therefore be to split the epicardial contour at the two landmarks and obtain two open, fixed-end sub-contours. This approach is extended to the endocardial contour by defining two pseudo-landmarks at minimum distance to the epicardial landmarks and split the contour in a similar way. It would be possible to optimise the endocardial given only one of the pseudo-landmarks but the splitting is more convenient since it gives open fixed-end sub-contours, similar to the other three and the contour parameters needed for the optimisation can be kept constant throughout the whole procedure. Figure 6.5 shows the five different sub-contour types. We will refer to these sub-contour types as:

- Right epicardial
- Left epicardial
- Right endocardial
- Left endocardial
- Right ventricle (RV)

To clarify, the names are also marked on the figure.



**Figure 6.5:** Splitting of the cardiac shape to five contours.

### 6.3.3 Optimisation procedure

Before going into details about the optimisation procedure of the five sub-contour sets, a few things must be clarified. The problem we are faced with is to obtain a **global** point correspondence, i.e. for the set of **full** cardiac shapes. However, due to the reasons mentioned in last section, the full cardiac shape had to be split up into five sub-contours. Optimising each of the five sub-contour sets completely independent of the others would give a local minimum, i.e. it would not be optimal for the full cardiac shape. The aim is therefore to optimise each sub-contour set individually, **with influence from the full shape**.

The actual optimisation process of sliding points until the description length is minimised must be done separately for each sub-contour set due to the properties of the parameterisation function. However, it is possible to affect the value of the objective function (description length) by the full shape. The factors, which affect the description length are the (Procrustes) alignment of the shapes and the PCA. The solution closest to global optimum would therefore be obtained by aligning globally and doing the PCA globally.

However, it gave better results to do a global alignment followed by local centering, (not local normalisation) and local PCA. For this reason along with less computation time, this approach was the one used.

The following section describes the need for balancing the point density of the shapes when doing a Procrustes alignment on the full set of shapes during the optimisation procedure.

### 6.3.4 Balanced alignment

The Procrustes alignment performed on the set of full cardiac shapes during optimisation must be balanced in the sense of number of points describing each contour type.

The hierarchical parameterisation requires open contours to be described by  $2^Q+1$  points, where  $Q \in \mathbb{Z}^+$ . This means that the individual (local) contour being optimised has  $N_{local} = 2^Q+1$  points. In order to balance the alignment, weighting with respect to arclength is carried out. This means that contour set  $i$ , ( $i \neq local$ ) will have  $N_i$  equidistant points, where:

$$N_i = round \left( N_{local} \cdot \frac{AL_i}{AL_{local}} \right) \quad (6.30)$$

where  $AL_i$  is the mean arclength of contour set  $i$  and  $AL_{local}$  is the mean arclength of the local contour set. The mean arclength is approximated by

$$AL_i = \frac{1}{n_s} \sum_{k=1}^{n_s} \sum_{j=1}^{n_l-1} \|s_k(t_{j+1}) - s_k(t_j)\|, \quad i = 1 \dots, 5. \quad (6.31)$$

where  $s_k(t_j)$  denotes the  $j$ 'th landmark of shape  $k$ .  $N$  is recalculated for each of the five optimisations.

### 6.3.5 Computation time and implementation and issues

Now, a brief discussion of the implementation and computation times for the optimisation is given. This is discussed here in order to explain the need for an approximation to the full MDL solution.

The MDL adjustments for the cardiac data were implemented in MATLAB as an extension to Thodbergs MDL MATLAB code [68]. The basic code had been tested for only small data sets of up to 40 shapes.

With the large amount of data, 2000 examples for each of the five sets of sub-contours, the code becomes considerably slow and impractical to run. In fact this was tested and had been running for three weeks when it was stopped without results.

Parallelisation could give a speed-up of the code but it is considerably complicated (or impossible) in MATLAB and was therefore not considered as an option.

By noticing that the most shape variation occurs between patients and between slices, the approach is to generate a **non-temporal set** by selecting the last frame of each slice sequence for the MDL optimisation. This reduces each contour set to 40 examples which leads to an acceptable computation time.

Attempts were made to feed this result to the full optimisation of 2000 shapes as a good initial guess, but this did not give satisfactory improvement.

Consequently, the optimisation is performed on the non-temporal set and the parameterisation functions obtained for each shape are assumed to be constant throughout the corresponding slice sequence. This is one of the issues discussed in the following section.

### 6.3.6 Post processing

The output from MDL optimisation on the non-temporal set is a set of parameterisation functions; one for each shape (one time frame from each slice sequence) of each of the five sets. These parameterisation functions are applied to the remaining frames of the relevant slice sequence deriving landmarks for the examples not included in the optimisation. This procedure is clarified in Table 6.2.

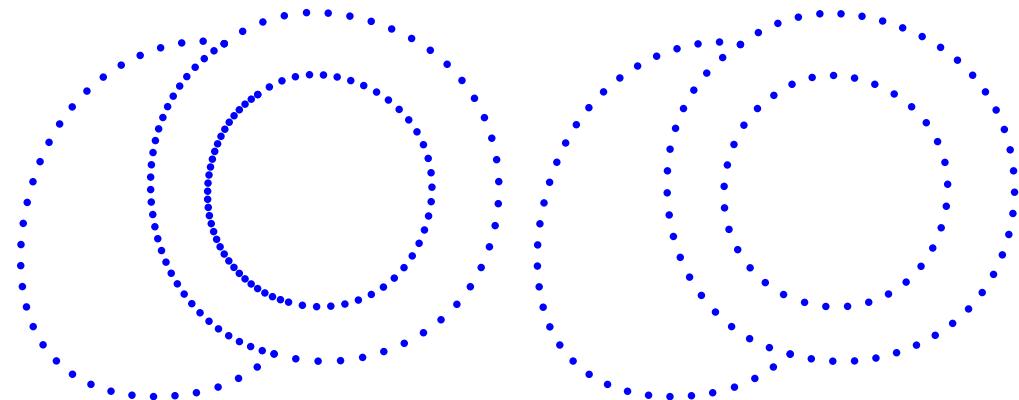
**Table 6.2:** Derivation of landmarks of full set from parameterisation functions of non-temporal set

1.	for all sub-contour sets
2.	for all patients
3.	for all slices
4.	for all time frames
5.	Derive shape landmarks by parameterising shape $s_{\text{patient},\text{slice},\text{frame}}$ by $\psi_{\text{patient},\text{slice}}$
6.	end frame loop
7.	end slice loop
8.	end patient loop
9.	end sub-contour loop

Each of the full (temporal) sub-contour sets are now described by  $2^Q + 1$  landmarks. Feeding this directly to AAM would give unequal weighting on different regions (sub-contours) of the cardiac shape. Consequently, balancing similar to the one described in Section 6.3.4 is performed. The estimation of the number of points to describe each sub-contour type is similar to Equation 6.30, except that the local number of points,  $N_{local}$  is not restricted to  $2^Q + 1$ . This only needs to be calculated once, where the (local) reference contour and the corresponding number of points,  $N_{local}$  are chosen freely.

The new solution is calculated by resampling the given number of points from the relevant MDL parameterisation function. The actual shape points are then extracted from the resampled parameterisation functions.

To illustrate the need for this post processing, Figure 6.6 shows the combined Procrustes mean shape before and after balancing.



**Figure 6.6:** Left: Procrustes mean shape, before resampling. Right: Procrustes mean shape, after resampling with respect to the mean arclength of each sub-contour.

### 6.3.7 Summary of cardiac MDL adjustments

The procedure for applying MDL shape modelling on cardiac contours is given in Table 6.3 as an extension to Thodbergs algorithm (Table 6.1). The adjustments made for the cardiac contours are bold-lettered.

**Table 6.3:** Pseudo-code for Thodbergs MDL optimisation algorithm [68] with adjustments for cardiac shape contours added.

```

1. Reduce the temporal set to a non-temporal set
2. Split shapes into 5 sub-contour sets
3. Loop over 5 sub-contour sets
4.   Determine #points for each sub-contour set by Eq. 6.30
5.   Do equidistant resampling using #points from 4.
6.   Loop over passes
7.     Loop over nodes
8.       Loop over 5 steps
9.         Loop over shape examples
10.        Loop over ± step
11.          Probe  $\tau(\text{node}) = \tau(\text{node}) \pm \text{stepsize of example}$ 
12.          Extract marks from parameterisation
13.          Center and normalise full set
14.          Do Procrustes of full set
15.          Center local set
16.          Do PCA of local set
17.          Compute DL from PCA
18.          If  $\text{DL} < \text{MDL}$ , set  $\text{MDL} \leftarrow \text{DL}$  and break loop
19.          Undo  $\tau(\text{node})$  change
20.        End of ± step loop
21.      End of example loop
22.      Adjust stepsize
23.    End of step loop
24.  End of node loop
25. End of passes loop
26. Derive landmarks for remaining frames of each slice sequence
     using optimal parameterisation functions.
27. End of sub-contour set loop
28. Redefine #landmarks for each sub-contour set solution by Eq. 6.30
29. Resample parameterisation functions using #landmarks from 28 and
     obtain new shape landmarks
30. Combine solutions and obtain the full training set of cardiac shapes

```

## 6.4 Summary and next steps

This chapter has covered the issues of semi-automatic landmarking. MDL shape modelling in general was presented followed by a discussion of the adjustments made for the cardiac data to fit the MDL framework. Computation time and implementation issues resulted in the approximation of reducing the full data set to a non-temporal set. Nevertheless, a framework for automatically creating landmarks on a set of cardiac shapes by MDL shape modelling has been provided. This results in a training set of shapes required for the AAM based registration method discussed in next chapter.



## Chapter 7

# AAM based registration method

Given the semi-automatically generated training set from last two chapters, shape and texture deformations of the myocardial perfusion data can now be learned by an Active Appearance Model (AAM) and unseen instances of the data can be synthesised and registered. This section deals with these topics.

First, a general discussion of AAMs is provided. Secondly modelling of the MRI perfusion data is discussed followed by the resulting modelling choice, clustering with respect to texture. Modelling of object interfaces and multi-slice shape modelling are the subject of next two sections and finally, model constraints and initialisation techniques are discussed.

### 7.1 Basic concepts of AAM

AAMs [18, 29] were introduced as a learning-based method for registration and interpretation of face images. By being a generic approach for image registration and image interpretation, medical applications were soon to follow (refer to [65] for a summary of medical AAM applications). The AAMs are built on the basis of a training set with the important property of landmark correspondences. From this set, both shape and texture properties are learned. From these quantities new images similar to the training set can be generated. Given a large enough training set it should be possible to generate any image of normal anatomy. Here, the term "normal anatomy" refers to that the model is general enough to generate any "legal" (normal) example of the class of images and furthermore it is specific enough not to generate any "illegal" instances of the class. Consequently AAMs can be fitted to unseen images, providing image registration and interpretation.

An appearance model is built by combining a model of texture variations and a

model of shape variations. After applying Procrustes analysis to align the shapes to a normalised mean shape, a statistical shape model is built by PCA. Subsequently each texture example is warped to match its landmarks to the landmarks of the mean shape. This results in a shape free patch from which a texture vector is sampled. After normalising the texture vector a statistical model of texture variation is built using PCA. Let  $\mathbf{s}$  and  $\mathbf{t}$  denote a synthesised shape and texture and let  $\bar{\mathbf{s}}$  and  $\bar{\mathbf{t}}$  denote the corresponding sample means. New instances of shape and texture are now generated by adjusting the principal component scores,  $\mathbf{b}_s$  and  $\mathbf{b}_t$  in

$$\mathbf{s} = \bar{\mathbf{s}} + \Phi_s \mathbf{b}_s , \quad \mathbf{t} = \bar{\mathbf{t}} + \Phi_t \mathbf{b}_t, \quad (7.1)$$

where  $\Phi_s$  and  $\Phi_t$  are eigenvectors of the shape and texture covariance matrices estimated from the training set. To obtain a combined shape and texture parameterisation,  $\mathbf{c}$ , the values of  $\mathbf{b}_s$  and  $\mathbf{b}_t$  over the training set are combined into

$$\mathbf{b} = \begin{bmatrix} \mathbf{W}_s \mathbf{b}_s \\ \mathbf{b}_t \end{bmatrix} = \begin{bmatrix} \mathbf{W}_s \Phi_s^T (\mathbf{s} - \bar{\mathbf{s}}) \\ \Phi_t^T (\mathbf{t} - \bar{\mathbf{t}}) \end{bmatrix}. \quad (7.2)$$

The shape and texture parameters have different units, i.e.  $\mathbf{b}_s$  is in units of pixel distances and  $\mathbf{b}_t$  in units of intensity. This is corrected for by choosing appropriate shape parameter weights stored in the diagonal matrix  $\mathbf{W}_s$  (see e.g. [20]).

To recover any correlation between shape and texture the two eigenspaces are usually coupled through a third PCA,

$$\mathbf{b} = \Phi_c \mathbf{c} = \begin{bmatrix} \Phi_{c,s} \\ \Phi_{c,t} \end{bmatrix} \mathbf{c}, \quad (7.3)$$

obtaining the combined appearance model parameters,  $\mathbf{c}$ , that generate new object instances (in the model space) by

$$\mathbf{s} = \bar{\mathbf{s}} + \Phi_s \mathbf{W}_s^{-1} \Phi_{c,s} \mathbf{c} , \quad \mathbf{t} = \bar{\mathbf{t}} + \Phi_t \Phi_{c,t} \mathbf{c}. \quad (7.4)$$

In order to generate new object instances in the image space, the pixel intensities in the texture vector  $\mathbf{t}$  are sampled into the geometry of the shape  $\mathbf{s}$  and the object is warped into the image space by pose parameters  $\mathbf{p} = [t_x \ t_y \ s \ \theta]^T$  where  $t_x$ ,  $t_y$  and  $\theta$  denote in-plane translation and rotation, and  $s$  denotes the shape size. The image warping method requires a Delaunay triangulation to recover the texture information between the landmarks of  $\mathbf{s}$ .

The aim is now to minimise the residual vector, i.e. the difference between the model and the true underlying image,

$$\mathbf{r}(\mathbf{q}) = \mathbf{t}_{image} - \mathbf{t} , \quad (7.5)$$

where  $\mathbf{q}$  are the parameters of the model,  $\mathbf{q}^T = [\mathbf{c}^T \ \mathbf{p}^T]$ . A first order Taylor expansion of Equation 7.5 gives

$$\mathbf{r}(\mathbf{q} + \delta\mathbf{q}) = \mathbf{r}(\mathbf{q}) + \frac{\delta\mathbf{r}}{\delta\mathbf{q}}\delta\mathbf{q} . \quad (7.6)$$

The goal is to find the parameter update  $\delta\mathbf{q}$  which minimises the residuals in the least-squares sense

$$\operatorname{argmin}_{\delta\mathbf{q}} \|\mathbf{r}(\mathbf{q} + \delta\mathbf{q})\|_2^2 . \quad (7.7)$$

Setting the right side of Equation 7.6 equal to zero, the following solution – the Jacobian estimate is obtained.

$$\delta\mathbf{q} = -\mathbf{R}\mathbf{r}(\mathbf{q}), \quad \mathbf{R} = \left( \frac{\delta\mathbf{r}^\top}{\delta\mathbf{q}} \frac{\delta\mathbf{r}}{\delta\mathbf{q}} \right)^{-1} \delta\mathbf{r}^\top \delta\mathbf{q} . \quad (7.8)$$

This estimate forms the basis for an iterative parameter updating scheme which should converge in the optimal image-to-model fit. The  $\mathbf{R}$  matrix is approximated initially from the training set and is kept fixed throughout the procedure.

For further details on AAMs refer to [18, 19, 20].

## 7.2 Modelling of perfusion MRI time-series

Since perfusion MRI sequences differ in structure from the single-image oriented AAMs, this section will discuss the issues of data modelling.

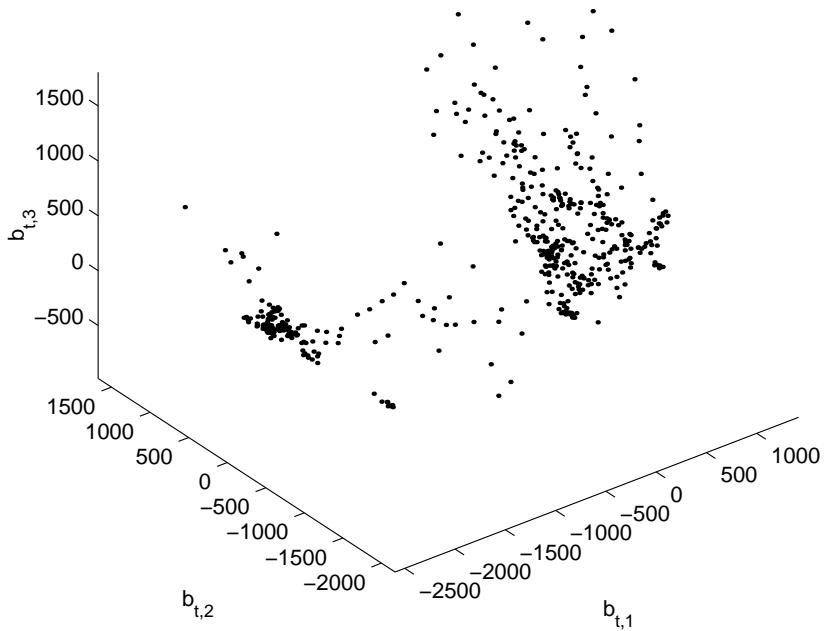
First, one should recognise that the signal variation of the left ventricle (LV) and the right ventricle (RV) is very small prior to contrast arrival. Due to this lack of image contrast, the standardisation of texture vectors normally used in AAMs would result in severe amplification of scanner noise. Hence, only the texture mean is removed in the following pre-processing of image texture vectors.

Deviations from the assumptions in standard AAMs are not only encountered with respect to texture normalisation. The relationship between shape and texture also differs. It is safe to assume that the process generating shape variation remains stationary throughout a sequence, contrary to the highly non-stationary texture process. Consequently, shape and texture are treated independently by removing the combined PCA, i.e.  $\Phi_c = \mathbf{I}$ . Treating each perfusion sequence as one observation (as done on cardiac cine MRI in [48]) is not feasible due to the random fluctuations in pose and shape induced by the variation sources mentioned in Section 2.2; whereof respiration dominates.

However, joint modelling of texture – decoupled from changes in shape and pose – for a complete sequence is possible by introducing an annotation of the temporal

relationship of each sequence frame. However, this may not be desirable, as only a few proper temporal landmarks exists (during the actual bolus passage). Further, building sequence models would require far more training examples to model this behaviour properly, compared to a simpler and less constrained frame-based model. Consequently, given the low number of subjects, each frame in a sequence will be treated as an observation.

Circumventing the need for large training sets unfortunately violates a basic assumption in AAMs, namely that the variation in texture is well modelled by a single multivariate Gaussian model. Due to the radical changes in intensity during contrast passage and uptake this is clearly not the case. On a coarse level we can split the sequence into pre-contrast arrival, contrast agent entering the RV, LV, and the myocardium.



**Figure 7.1:** First, second and third principal component of 500 texture vectors from ten perfusion sequences.

Since the texture parameters of the texture model shown in Equation 7.1 are ranked by their variance over the training set, this alleged clustered behaviour should be confirmed by visual inspection of a few highly-ranked parameters from  $\mathbf{b}_t$ . Figure 7.1 shows the three most significant texture parameters,  $\{b_{t,i}\}_1^3$ , from a texture model built from the ten available perfusion sequences. This figure exhibits a pronounced clustering and thus verifies our concerns. Modelling this

multimodal distribution with a multivariate Gaussian is clearly not justified and gives rise to several problems. Most problematic is that since the resulting model is not specific to the given problem it can easily generate textures that are not plausible to occur during a perfusion bolus passage. This would in turn lead to potential false positives during the model-to-image matching and will be dealt with in the following section.

### 7.3 Adding cluster awareness

To model the distribution of textures we propose an unsupervised learning approach that models texture variation using an ensemble of linear subspaces in lieu of the unimodal linear model employed in AAMs. Alternatively to clustering, these subspaces could have been given by an operator, which identifies different phases of each perfusion sequence. However, to reduce firstly the tedious burden of training set generation and secondly inter-and intra-observer variability, supervised learning was rejected. Further, we like to leave the option open to evaluate different ensemble sizes in the future when more training data becomes available. This will obviously be very tedious in the case of manual labelling.

Although the machine learning literature offers an abundance of classification methods, it is generally agreed upon that no *silver bullet* exists. The approach used in this study is a  $k$ -means classification [32] combined with a Monte Carlo simulation scheme where several classifications are carried out, based on different initial random class centres. The final classification is chosen using a minimax criterion, where the classification having the smallest maximum distance to the nearest class centre is chosen.

From the resulting classification a set of linear texture subspaces,  $\{\Phi_{t,i}\}_{i=1}^{n_c}$ , is obtained directly by  $n_c$  separate texture PCAs. A corresponding set of model parameter update matrices,  $\{\mathbf{R}_i\}_{i=1}^{n_c}$ , is obtained following the procedure in [19] using a displacement scheme specified in [65].

As texture changes in a sequence are deemed to be unrelated to shape changes, building a joint shape model,  $\Phi_s$ , from all frames in all sequences will yield the best estimate of inter- and intra-subject shape variability. This composition of a single shape model and an ensemble of texture models with associated parameter update matrices is noted as a *Cluster-aware Active Appearance Model* (CAAM). A CAAM can thus be applied to any domain problem that justifies uncoupled shape and texture eigenspaces and requires piecewise-linear modelling of texture variability.

Fitting a CAAM to unseen images now involves choosing the appropriate texture subspace. As a reasonable choice for  $n_c$  is typically fairly limited, model selection is here performed by exhaustively trying all models and selecting the model producing the best model-to-image fit, subject to a set of constraints addressed in

Section 7.6. To increase performance during model fitting, model selection could be accomplished by a classification of the texture vector into the set of training classes. However, this has not been tried in the current work.

To choose  $n_c$ , prior knowledge can be employed. However, being an optimisation problem in one positive integer variable, a data-driven method should be preferred where  $n_c$  is estimated using cross-validation on the training set. However, observe that the number of available training examples for model estimation within a single class is inversely proportional with  $n_c$ . Due to this fact, we have fixed  $n_c$  to five classes for all experiments as a reasonable compromise, since conclusions regarding the optimal number of classes would remain fragile, due to the limited number of training subjects available.

## 7.4 Modelling object interfaces

To add notion about the interface between the heart and the surrounding tissue et cetera, intensity samples in the proximity of the heart are included in the texture model.

This is carried out by sampling landmark normals relative to the current shape size, similar to the approach by Active Shape Models, [21]. These normals are denoted *whiskers*. In our application, this will include the LV/lung interface and LV-RV/abdomen interface into the texture models and thus require these to be present in the unseen image to provide a good model-to-image fit. In all of the subsequent experiments whisker samples are weighted so that they constitute one third of the texture variation.

This approach has earlier been shown to have a positive, and significant, impact on the registration accuracy, in the case study on corpus callosum registration in brain MRI, [64].

## 7.5 Multi-slice shape modelling

Up to this point an observation has been associated to a single image. However, cardiac perfusion MRI is typically acquired as multi-slice images in a spatial arrangement. This enables regional assessment of the perfusion mechanism both in plane as well as through plane. Modelling such slice images independently will per se disregard any spatial and/or intensity coherence between slices. Conversely, a joint modelling of each set of slices will provide a constrained basis, in which slices with well-defined appearance will restrict slices with diffuse appearance from diverging.

Coupled AAMs were first explored in [22] for interpretation of multi-view face

images. Later, time-sequences of cardiac MRI were coupled [48], an approach which was also pursued in [44, 59, 11, 12, 13]. Recently, long and short axis MRI, and end-diastolic and end-systolic angiograms were fused by a similar approach by [45] and [52], respectively.

How coupled AAMs should be constructed is application specific. Here, the following approach is taken: Let a shape consisting of  $N$  points in two dimensions be defined by a  $2N$  vector, as represented by Equation 4.1.

$$\mathbf{s} = [x_1 \ y_1 \ x_2 \ y_2 \ \dots \ x_N \ y_N]^T. \quad (7.9)$$

A multi-slice observation consisting of a stack of  $C$  slices is then composed of the set of shapes;  $\{\mathbf{s}_i\}_{i=1}^C$  by simple concatenation of the unmodified slice image shape coordinates prior to normalisation et cetera,

$$\mathbf{s}_{frame} = [s_1^T \ s_2^T \ \dots \ s_C^T]^T. \quad (7.10)$$

Any subsequent shape analysis is left unchanged. Image sampling is carried out using the appropriate slice for each related subpart of the combined shape vector. This produces a model where inter-slice pose relations are modelled by the shape changes, which is clearly desirable for the application in question. Should inter-slice differences in pose not be constrained; then slice concatenation should be applied to the Procrustes tangent space shape coordinates instead. This will thus introduce the need for  $(C - 1) \times 4$  extra pose parameters into the model (in the 2D case).

## 7.6 Estimating and enforcing pose and shape priors

The fact that changes in pose and shape are unrelated to the change of texture is highly useful for initialising and constraining the model fitting process in each frame. Further, it can validate the final registration results as well as the intermediate results for each texture class. Consequently, if it is possible to obtain reliable estimates of the shape and pose variation in a subpart of the sequence these can be used as constraints in the remaining part of that sequence. This is the case in the latter part of a bolus passage where the contrast agent has been washed out of the RV and LV, only leaving the subtle changes stemming from the perfusion mechanism in the myocardium. Hence, we propose to estimate prior distributions of pose and shape from the latter part of a perfusion sequence of  $P$  frames.

Let  $\kappa, \gamma, D_{max}$  denote a set of user-selectable, dimensionless, constants controlling the influence of the priors. Then, let  $\Sigma_p$  denote the dispersion matrix of the sequence pose parameters and let  $\sigma$  denote the standard deviations of the sequence shape parameters. Further, let  $\mathbf{F}_t$  denote the  $t$ -th frame. Let the set of frames  $\{\mathbf{F}_t\}_{t=1}^{S-1}$  denote the unstable period, and the frames  $\{\mathbf{F}_t\}_{t=S}^P$  denote the stable

period. A formal scheme for exploiting these priors can then be formulated as shown in Table 7.1.

**Table 7.1:** Sequence prior augmented CAAM search

	Require: $S, \kappa, \gamma, D_{max}, \Sigma_p$ and $\sigma$
1.	$\bar{\mathbf{p}} = \text{initialisation}$
2.	$\bar{\mathbf{b}}_s = \text{initialisation}$
3.	for $t = P$ down to $S$
4.	$\{\mathbf{p}_t, \mathbf{b}_{s,t}\} = \text{CAAM search started at } \{\bar{\mathbf{p}}, \bar{\mathbf{b}}_s\} \text{ in } \mathbf{F}_t$
5.	$\bar{\mathbf{p}} = \frac{1}{P-t+1} \sum_{j=t}^P \mathbf{p}_j$
6.	$\bar{\mathbf{b}}_s = \frac{1}{P-t+1} \sum_{j=t}^P \mathbf{b}_{s,j}$
7.	end for
8.	for $t = S - 1$ down to 1
9.	$\{\mathbf{p}_t, \mathbf{b}_{s,t}\} = \text{Constrained CAAM search started at } \{\bar{\mathbf{p}}, \bar{\mathbf{b}}_s\} \text{ in } \mathbf{F}_t$ (constrain using $\kappa, \gamma, D_{max}, \Sigma_p, \bar{\mathbf{b}}_s, \sigma$ )
10.	end for

During CAAM search in the unstable period, pose and shape priors are used to stabilise parameter updates by limiting the maximal update step. To simplify notation, the time index  $t$  is omitted from this point on. Let  $\Sigma_{p(ij)}$  denote the element in the  $i$ -th row and  $j$ -th column of  $\Sigma_p$ , and let  $p_i$  denote the  $i$ -th element of  $\mathbf{p}$ . Pose parameter updates can now be constrained using the following simple clamping approach:

$$\delta p_i = \begin{cases} \text{sign}(\delta p_i) \kappa \sqrt{\Sigma_{p(ii)}} & \text{if } |\delta p_i| > \kappa \sqrt{\Sigma_{p(ii)}} \\ \delta p_j & \text{otherwise.} \end{cases} \quad (7.11)$$

The constant  $\kappa$  acts thus as a clamping constant given in units of standard deviations of pose variation as estimated from the training sequences. In all experiments  $\kappa = 0.5$  was used. As an alternative to this hyper cuboid constraint we could use a projection onto the hyper ellipsoid given by an eigenanalysis of  $\Sigma_p$ .

Likewise, we also exploit the prior knowledge of sequence shape variation,  $\sigma$ , (as obtained from the training set) in a similar clamping approach,

$$b_{s,i} = \begin{cases} \bar{b}_{s,i} + \text{sign}(b_{s,i} - \bar{b}_{s,i}) \gamma \sigma_i & \text{if } |b_{s,i} - \bar{b}_{s,i}| > \gamma \sigma_i \\ b_{s,i} & \text{otherwise,} \end{cases} \quad (7.12)$$

where  $\gamma$  denotes the maximally accepted distance from the mean in units of standard deviations. In all experiments  $\gamma = 3$  was used.

After ended CAAM search in the unstable period, pose prior enforcement is determined by testing the Mahalanobis distance to the pose distribution:

$$\{\mathbf{p}, \mathbf{b}_s\} = \begin{cases} \{\bar{\mathbf{p}}, \bar{\mathbf{b}}_s\} & \text{if } D_{max}^2 < (\mathbf{p} - \bar{\mathbf{p}})^\top \Sigma_p^{-1} (\mathbf{p} - \bar{\mathbf{p}}) \\ \{\mathbf{p}, \mathbf{b}_s\} & \text{otherwise} \end{cases} \quad (7.13)$$

Under the assumption of normally distributed pose parameters, it was decided to use  $D_{max} = 3$ . Hence, implausible pose solutions are discarded and replaced with the maximum likelihood of the prior; the mean configuration. Such solutions are flagged and the system operator can be prompted for manual assistance. This pose prior enforcement is also employed during texture model selection so that solutions failing the above condition are excluded from the competition of being the selected for the current frame. This step is essential, since highly deviating pose configurations easily can produce model-to-image fits that will outperform the true positives.

## 7.7 Model initialisation

The initial pose, shape and texture parameter configuration of the CAAM, in frame  $P$ , is determined by exploiting the convergence radius of the AAM search. Within this radius the AAM will converge to a plausible fit. Parameters with variation outside this radius over the training set are semi-exhaustively searched, with grid spacing less than twice the radius, by starting the AAM search at each grid point. Each search result represents a potential initialisation candidate. To add robustness this is implemented in a candidate scheme, where several candidates compete in an evolutionary process to become the initial configuration. See [62] for the details of this algorithm.

Similar to the normal CAAM search, initialisation is performed exhaustively for all texture models and subsequently choosing the model providing the best fit.

## 7.8 Summary and next steps

This chapter has described a registration method based on AAM with extensions mainly related to the myocardial perfusion data. To cover the intensity differences across the perfusion sequences, clustering of texture vectors was introduced. Additionally, a coupled slice model was presented to utilise the relationship between slices. Now that per-pixel registered perfusion sequences can be obtained, perfusion assessment is the next step.



## Chapter 8

# Perfusion assessment

The actual goal of the image registration discussed in previous chapters is to assess myocardial perfusion (blood flow to myocardium). Perfusion analysis of MR perfusion images can be carried out in three steps: *Qualitatively, semi-quantitatively* and *quantitatively* [49].

The qualitative assessment can be done without image registration, since it is based on inspections of contrast between myocardial regions in the original images. For the other two types of perfusion study, fully registered image sequences must be available.

The semi-quantitative analysis relies on the so called signal-intensity (SI) curves, which are obtained by plotting intensity versus time frames in the myocardium at corresponding locations, pixel-wise or region-wise. See e.g. [14, 16, 39, 47, 50, 53, 61].

The fully quantitative analysis, which results in an absolute measure of myocardial perfusion in  $\text{ml/g/min}^1$ , is based on treating the SI-curves as a response for the injected contrast bolus, see e.g. [40, 43].

Table 8.1 summarises the three steps to cardiac perfusion analysis and gives the clinical correlation to each of the steps [49].

An absolute quantitative perfusion analysis is a full study in itself and will not be carried out here. Instead semi-quantitative analysis of the previously registered images is performed providing a basis for a full quantification. This is usually done independently for each slice sequence and will be the approach followed here.

---

<sup>1</sup>ml blood per gram of tissue per minute.

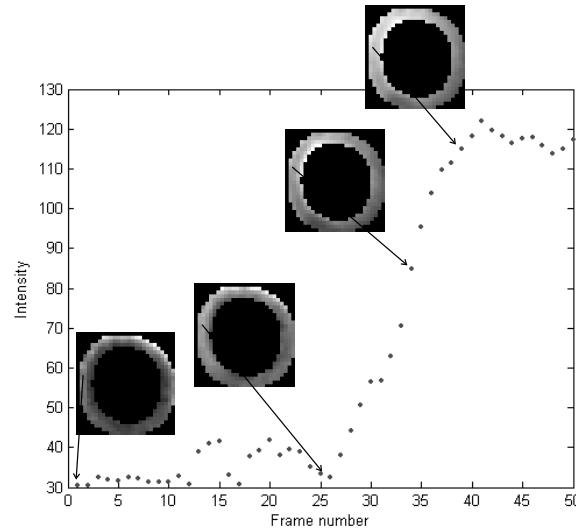
**Table 8.1:** Three steps of analysis of cardiac MR perfusion images.

Step	Assessment	Clinical correlation
Qualitative	Contrast between myocardial regions in original images	Localisation and extent of ischaemia or infarction.
Semi-quantitative	Properties of SI-curves	Relative differences between myocardial regions.
Quantitative	Absolute myocardial perfusion in ml/g/min.	Absolute differences between myocardial regions and between patients or studies.

## 8.1 Semi-quantitative perfusion assessment

Due to the inherent representation of texture vectors in AAMs, pose- and shape-compensated images are *directly* obtained by projecting each texture vector into the shape-free reference frame. Thereby a per-pixel correspondence over the complete perfusion sequence can be obtained and SI-curves can easily be generated.

To clarify, Figure 8.1 gives the relationship between an SI-curve and the corresponding positions within the myocardium image frames.

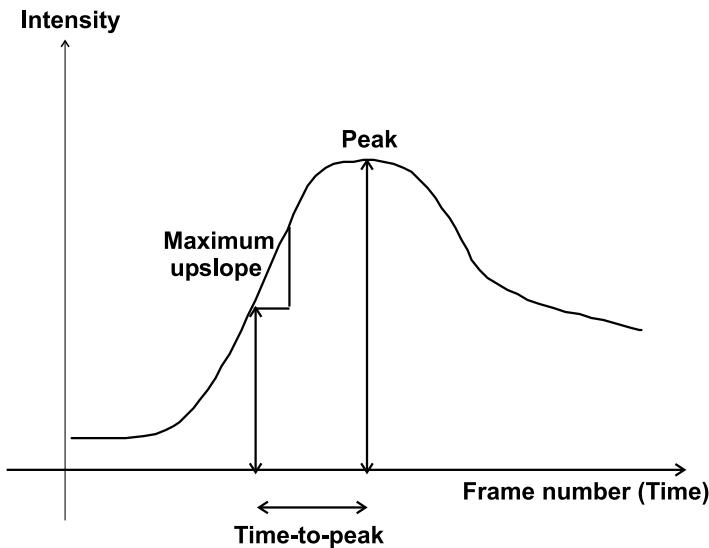


**Figure 8.1:** Relationship between the SI-curve of the myocardium and the corresponding positions within image frames.

The parameters used for the semi-quantification are derived from the SI-curve. The parameters calculated in this study and their relation to the SI-curve are the following:

- *Maximum up-slope*: Maximum intensity difference between two adjacent time frames
- *Peak*: Maximum intensity value after reaching the maximum up-slope
- *Time-to-peak*: Time from maximum up-slope to peak<sup>2</sup>

To illustrate, Figure 8.2 gives the relation of the parameters to an ideal SI-curve of the myocardium.



**Figure 8.2:** A theoretical SI-curve from a region or pixel of the myocardium. The relation of perfusion parameters to the curve is also illustrated.

The SI-curve in Figure 8.2 illustrates the bolus passage. That is, it shows how the contrast agent enters the myocardium, reaches a maximum intensity (the peak) and then washes out again.

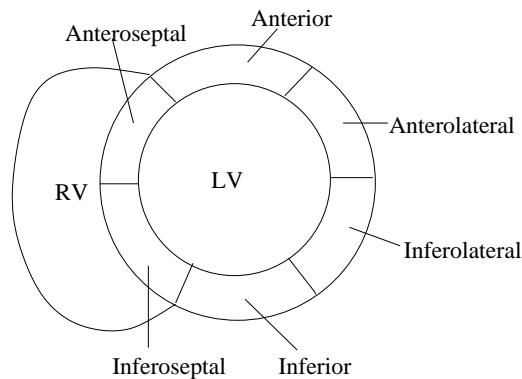
It has been shown that variations in the above mentioned parameters can be correlated reasonably with the blood flow. Thereby they can be used to detect ischemic segments of the myocardium.<sup>3</sup>

These regions can be assessed by so-called perfusion maps for each of the three parameters. The maps are generated by displaying the value of the parameter in each pixel position as intensity in a gray-scale image scaled with respect to the remaining values. This reveals relative differences between normal and ischemic segments. These differences may be small for the above mentioned parameters and therefore a parameter often assessed is the *perfusion reserve index*. It is calculated as the ratio of the maximum upslope for a stress scan and the maximum upslope for a rest scan [50]. Since the data of this study only covers rest scans this parameter could not be determined.

<sup>2</sup>Some approaches use the time from first intensity increase to peak.

<sup>3</sup>Refer to Chapter 1: Ischemic segments are regions of reduced blood flow.

We conclude this section by showing a schematic figure of the cardiac shape with regions of the myocardium labelled. This terminology will be used in the discussion of registration and perfusion results.



**Figure 8.3:** Regions of the myocardium.

## 8.2 Summary and next steps

This section has addressed the three different steps to absolute quantification of blood flow. In particular, the second step, semi-quantitative perfusion assessment performed in this project has been discussed. Now, the theoretical aspects of this thesis have been covered. In the next chapter, evaluation of the methods on the myocardial perfusion data will be carried out.

## Chapter 9

# Results and evaluation

This chapter presents experimental results for the data described in Section 2.3. This comprises results from MDL shape modelling, registration results from the AAM based method and finally example results from the semi-quantitative perfusion assessment.

### 9.1 Training set formation: Results from MDL

This section illustrates the automatic landmarking procedure on the extracted cardiac shapes from Chapter 5. A brief review of the relevant adjustments made for the myocardial perfusion data follows.

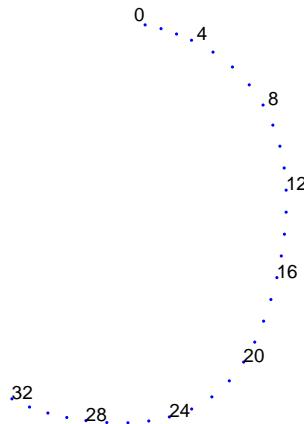
As addressed in Section 6.3.2, the cardiac shapes are split into five sub-contour sets (refer to Figure 6.5). Section 6.3.5 concluded that the full data set must be reduced due to poor computation time of the algorithm. As a result, five non-temporal data sets are defined, each including 40 sub-contours obtained from the last multi-slice frame of all patients. The MDL optimisation is carried out on each of the five non-temporal data sets with influence from the other four sets. The rough approximation used here is that the resulting parameterisation functions ( $\psi$ ) remain constant throughout each slice sequence. Landmarks can therefore be derived for the remaining time frames of a slice sequence using the corresponding optimal parameterisation function from the non-temporal set (refer to Table 6.2). Subsequently, the derived landmarks for the five sub-contours are resampled and merged to generate the full training set of cardiac shapes. To summarise, the procedure is listed below.

1. Split each cardiac shape into five sub-contours.
2. Reduce each sub-contour set to a non-temporal set using the last multi-slice frame of each patient.
3. Do MDL optimisation of each of the five non-temporal sets with influence from the remaining four sets.
4. Derive landmarks for each full sub-contour set by applying the optimal parameterisation functions from the non-temporal set to the remaining slice sequences.
5. Balance the point density between sub-contour sets by resampling the landmarks
6. Combine the sub-contour sets to generate a full training set.

The results presented here will predominantly concern step 3 of this procedure, i.e. the MDL optimisation itself. Additionally, inspection of the final result, the full training set generated in step 6 will be given.

### 9.1.1 MDL optimisation of the five sub-contour sets

For each of the five sub-contour optimisations, 33 equidistant landmarks were placed on each example as a start point. Marks number 0 and 32 are the anatomical or pseudo-landmarks which are kept fixed throughout the procedure. Nodes were placed at the first three levels in the hierarchy, i.e. marks number 16 (level 0), 8,24 (level 1), 4,12,20,28 (level 2) were defined as nodes. Figure 9.1 shows this labelling for the right endocardial sub-contour.

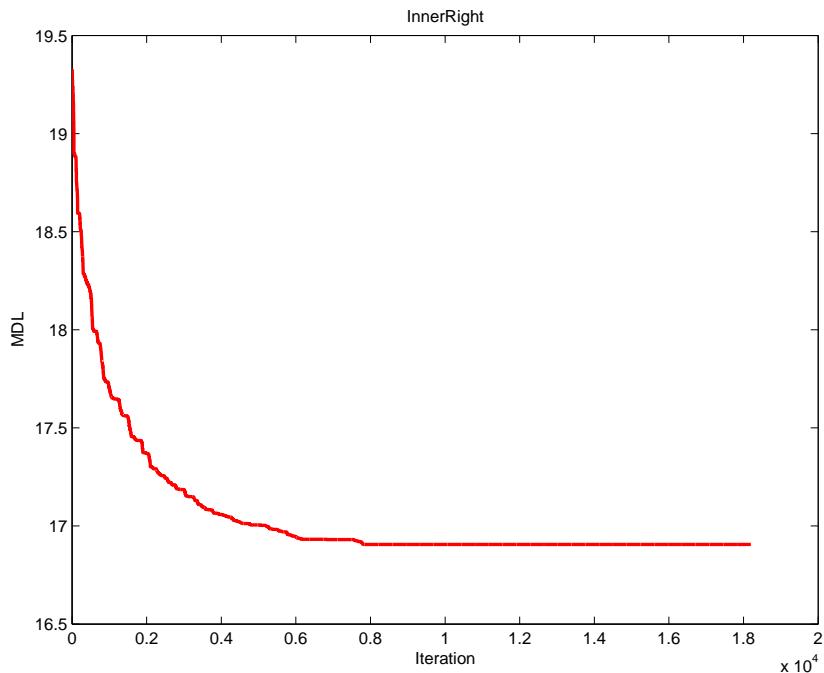


**Figure 9.1:** Equidistant landmarks on the right endocardial sub-contour. Points number 0 and 32 are fixed (anatomical landmarks). The remaining labelled points are nodes.

The tolerance parameter from Equation 6.29 was set to  $T = 0.05$  and  $\sigma_{cut} = \sqrt{\lambda_{cut}}$  was set to 0.003 in all optimisations. These are the default values in the code and have given good results in general.

Regarding the computation time, the optimisation of the five non-temporal sets took approximately 2 hours (on a 2.4 GHz Pentium PC) in total.

The development of the MDL optimisation will now be illustrated in two figures. Figure 9.2 shows the convergence of the MDL algorithm for the right endocardial sub-contour set. The plot shows clearly that the algorithm converges in a familiar way, that is, the value of the MDL decreases fast during the first few iterations and then slowly reaches a minimum. Similar plots were obtained for the remaining sub-contours.

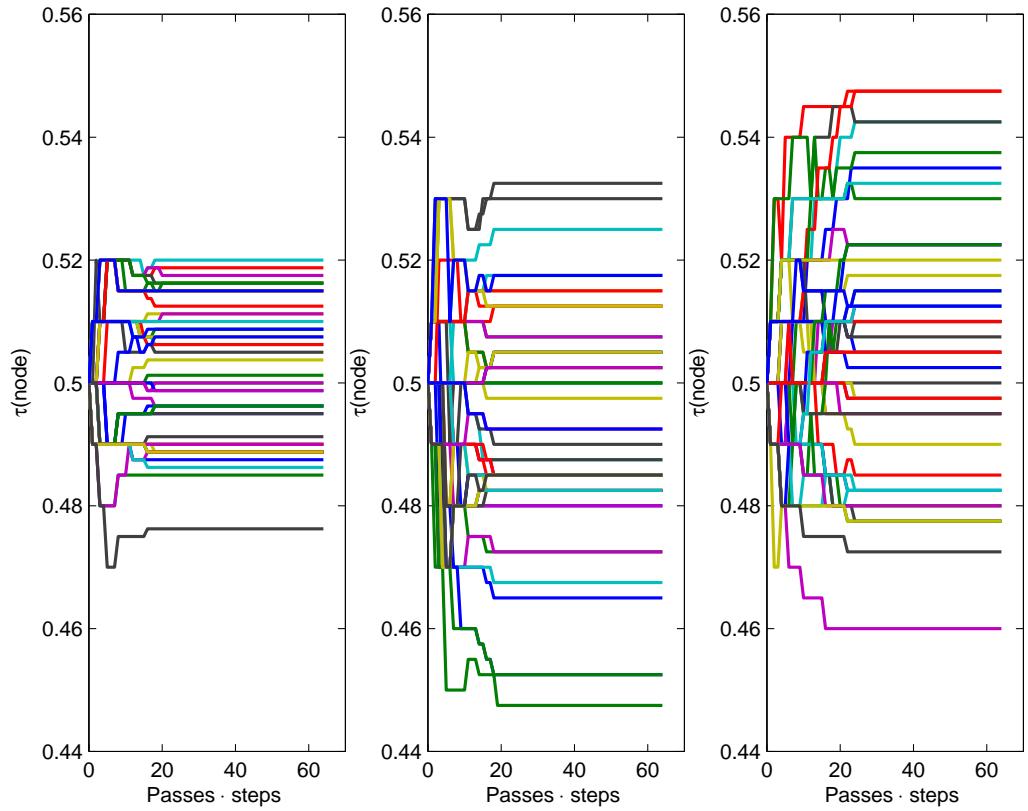


**Figure 9.2:** Convergence of MDL optimisation for the right endocardial sub-contour set. Note that the iteration number is composed of  $\# \text{passes} \cdot \# \text{nodes} \cdot \# \text{steps} \cdot \# \text{examples}$ .

Figure 9.3 shows the development of node positions for nodes on the first two levels in terms of  $\tau(\text{node})$  across the set of right endocardial sub-contours. The approximately symmetric structure of the plots around  $\tau(\text{node}) = 0.5$  indicates that the stabilising term (refer to Equation 6.29) is low. That is, the final average of node parameters is quite close to 0.5 for all nodes indicating only small changes from the equidistant parameterisation.

The development of the MDL optimisation for the right endocardial sub-contour set has now been addressed. The remainder of this section will illustrate quantitative and qualitative results from the MDL optimisations.

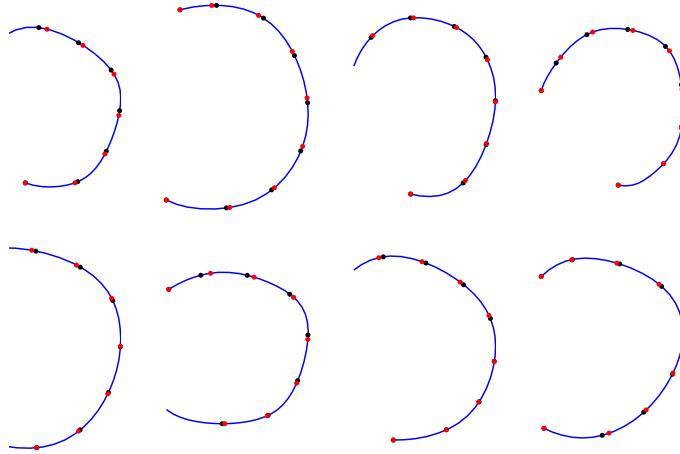
Figure 9.4 shows actual displacements of node positions on the right endocardial for patients 2 and 3. The plots confirm the indication of Figure 9.3 that the



**Figure 9.3:** Development of the positions of nodes on first two levels (nodes 16, 8 and 24 from left to right) for optimisation of right endocardial sub-contour set. Each node plot shows the value of  $\tau(\text{node})$  plotted versus  $\#\text{passes} \cdot \#\text{steps}$  for all examples (contours) in the set (different colours correspond to different examples).

optimal solution is not far from the equidistant solution. This is quite reasonable since neither anatomical nor mathematical landmarks are present on the right endocardial (the same applies for the other four sub-contours). Consequently, the equidistant solution is already a good initial guess.

To give an idea of how much improvement was accomplished in the five optimisations, Tables 9.1 and 9.2 are provided. They both give a quantitative measure of model compactness before and after optimisation. The first one by total model variance and the second one by the description length itself. The tables reveal that a considerable improvement is obtained for all contour sets.



**Figure 9.4:** Fixed points and node positions for the right endocardial sub-contour set. Equidistant (black) and MDL optimised (red) solution. First row: Patient 2, slice 1–4. Second row: Patient 3, slice 1–4.

**Table 9.1:** Comparison of total model variance for equidistant landmarking and MDL optimised landmarking. Percentage decrease is also shown.

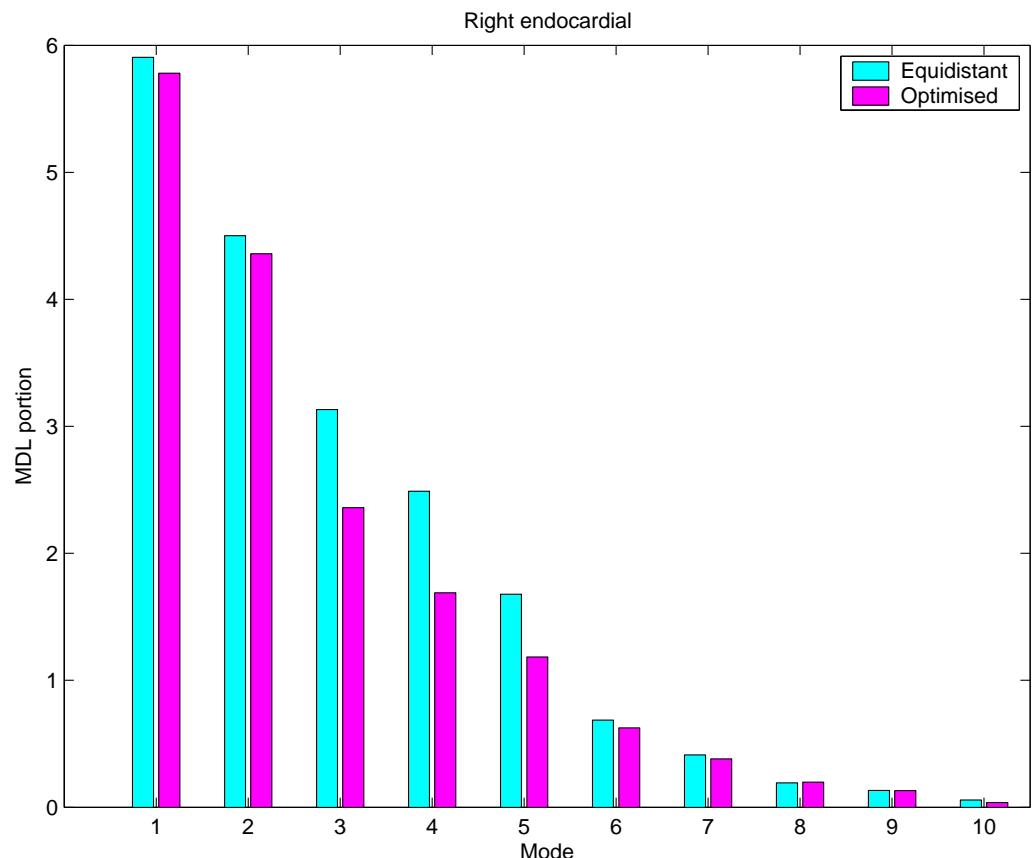
Sub-contour set	Equidistant $\times 10^3$	Optimised $\times 10^3$	Decrease
Right epicardial	2.60	1.80	28%
Left epicardial	0.96	0.76	21%
Right endocardial	1.70	1.40	15%
Left endocardial	0.47	0.37	22%
RV	4.0	3.60	10%

**Table 9.2:** Comparison of objective function (DL) for equidistant landmarking and optimised landmarking. Percentage decrease is also shown.

Sub-contour set	Equidistant	Optimised	Decrease
Right epicardial	24.3	22.1	10%
Left epicardial	14.7	13.4	10%
Right endocardial	19.3	17.0	14%
Left endocardial	11.0	9.80	11%
RV	31.0	27.9	11%

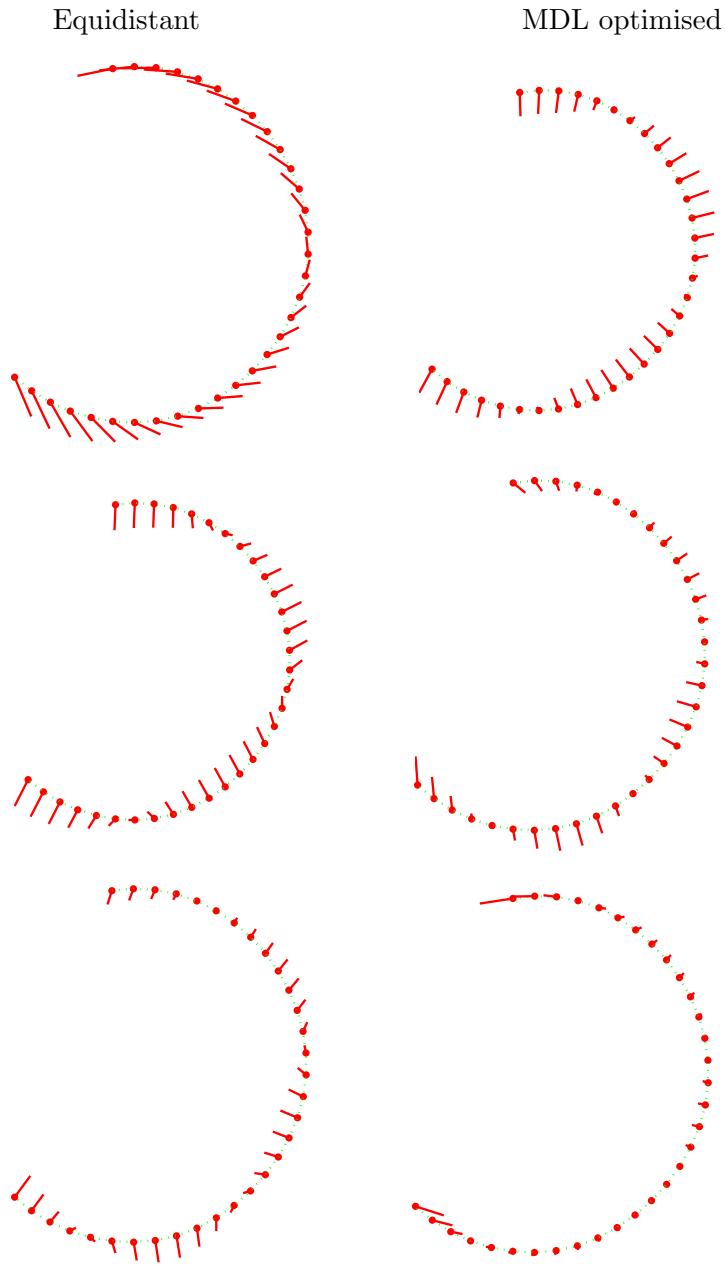
The results from Table 9.2 are now analysed in further details for the right endocardial sub-contour by considering the different contribution of each eigenvalue to the description length. Figure 9.5 gives the contribution of the first ten eigenvalues before and after optimisation for the right endocardial set. Figure 9.5 indicates that most improvement of the equidistant solution should be expected in modes 3,4 and 5. The reason that the improvement does not appear in earlier modes in general is that the first two modes mostly account for scaling factors. Those factors exist because normalisation was carried out with respect to the set of full

shapes (all sub-contours) but not within the individual sub-contour sets (refer to Section 6.3.3).



**Figure 9.5:** Contribution of the first 10 eigenvalues to the objective function. Comparison of equidistant landmarks and MDL optimised landmarks for the right endocardial set.

Figure 9.6 illustrates modes 3–5 for both the equidistant and optimised solution.



**Figure 9.6:** Comparison of principal modes 3, 4 and 5 (from top to bottom) before and after optimisation of the right endocardial contour set. Left: Equidistant landmarks, right: MDL optimised landmarks. The whiskers denote 5 standard deviations from mean.

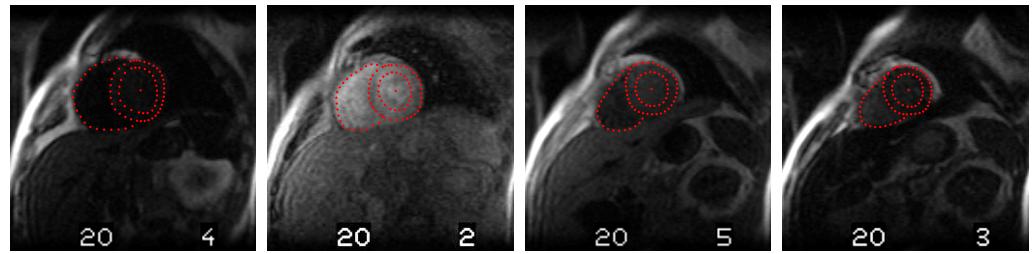
Figure 9.6 reveals a clear improvement for mode 3. The whiskers for the MDL optimised landmarks appear approximately orthogonal to the mean shape. This property is an indicator of a good shape model which in turn must be based on good point correspondence. This is not the case for mode 3 of the equidistant

model. The improvement is not as obvious for mode 4 and 5. However, it is noticed that modes 4 and 5 for the equidistant landmarks appear similar to (respectively) modes 3 and 4 for the MDL optimised landmarks. This indicates that a larger proportion of the total variance has been covered in the first four modes of the MDL solution compared to the first four modes of the equidistant solution.

### 9.1.2 Full training set

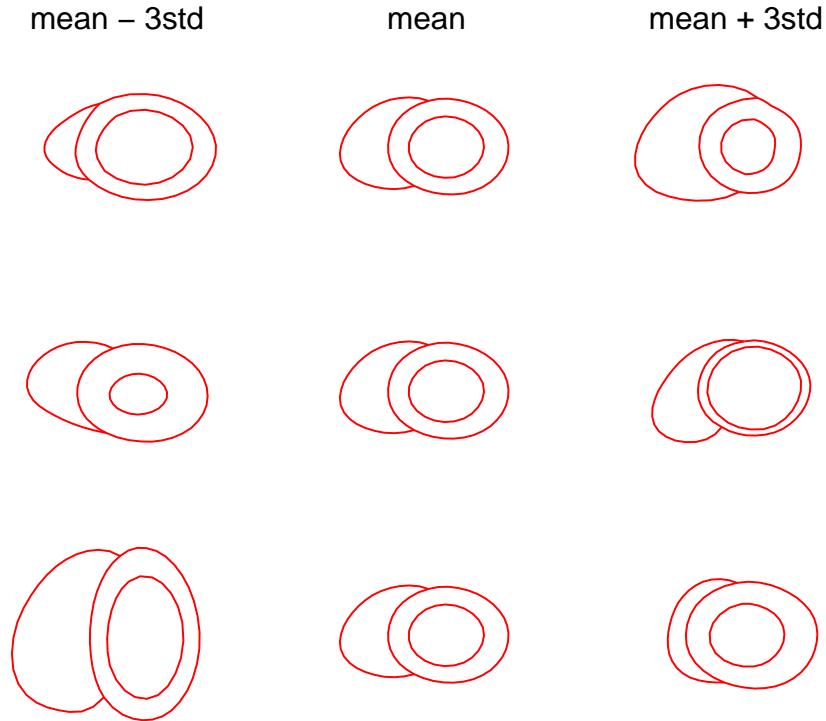
By performing steps 4-6 in the landmarking procedure stated in the beginning of this section a full training set is generated.

To give an example of how the derived landmarks appear on the image frames, we plot the derived landmarks for a multi-slice frame not included in the non-temporal set as seen in Figure 9.7.



**Figure 9.7:** Derived landmarks for the four slices of patient 1, frame 20.

Shape variation in this full training set was modelled by PCA. The resulting deformations for the first three principal modes are shown in Figure 9.8. Note that in this model, inter-slice variance is pooled (as discussed in Section 6.3.1). It is only intended to examine the results from MDL but not to base the multi-slice AAM model on. The figure reveals that the first mode describes variations in the RV and the second mode accounts for myocardium thickness. The third mode accounts for RV variations as well as variations in the myocardium shape.



**Figure 9.8:** First three principal modes (from top to bottom) illustrated as  $\pm 3$  standard deviations from mean shape of the full temporal set.

### 9.1.3 Discussion of MDL results

The MDL optimisation improved the landmarking of the non-temporal set substantially for all five sub-contour sets according to Tables 9.1 and 9.2. Still, the improvement is not as extensive as for the corpus callosum case presented in Section 6.2.4. This is quite reasonable since the cardiac sub-contour sets do not give the MDL optimisation much to "work with", i.e. neither anatomical nor mathematical landmarks are present as in the corpus callosum case. Consequently, an extreme improvement could not be expected.

Although the MDL optimisation gives good improvement for the non-temporal set, the derived landmarks on the full set are not much better than the initial equidistant landmarks. The approximation used by assuming constant parameterisation function throughout a slice sequence is quite crude and in fact it corresponds to the assumption that no shape variation occurs in the temporal dimension (of the full set).

Many attempts were made to avoid this approximation by making the optimisation run on the full set but without success. The largest factors which lead to poor computation time are that the pattern search algorithm is generally slow.

Additionally, it includes a lot of for-loops which implemented in MATLAB generally results in poor computation time. It would be interesting to compare this algorithm to the gradient based algorithm from [30]. These types of algorithms have better computational complexity than pattern search in general.

At this point, a full training set of the 2000 shapes has been generated semi-automatically. After concatenating images and shapes in the vertical direction to generate multi-slice frames as explained in Section 7.5, the multi-slice perfusion sequences can now be modelled by the AAM based method.

## 9.2 Image registration by AAM

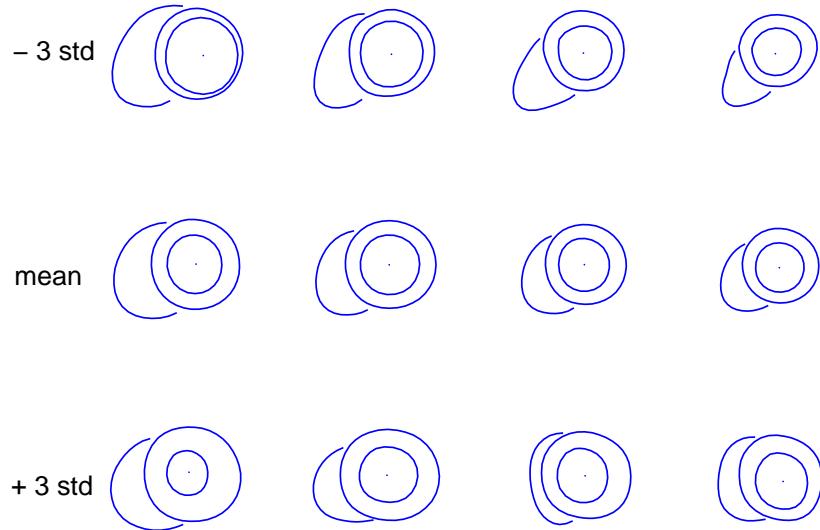
Given the slice-concatenated training set from the MDL optimisation, a validation of the AAM based registration method proposed in Chapter 7 was carried out. An extended C++ implementation of the AAM framework by Stegmann et al. [65] was executed on a 2.4 GHz Pentium PC. The total number of landmarks for each multi-slice frame in the training set was 332, including pseudo landmarks at the LV centre (explained later in this section).

Firstly, this section gives illustration of model training of the multi-slice set and secondly, results from a leave-one-out cross validation are presented. Eventually, a discussion of the registration results is provided.

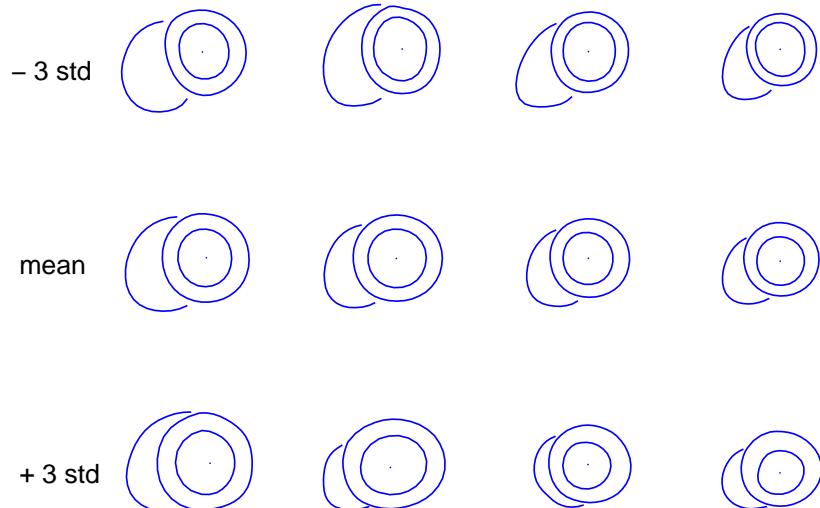
### 9.2.1 Model training

Recall from Sections 7.2 and 7.3 that texture variation is divided into  $n_c$  models where  $n_c$  is the number of classes used in clustering of texture vectors. However, shape variation is modelled using all 500 (multi-slice) examples in the training set.

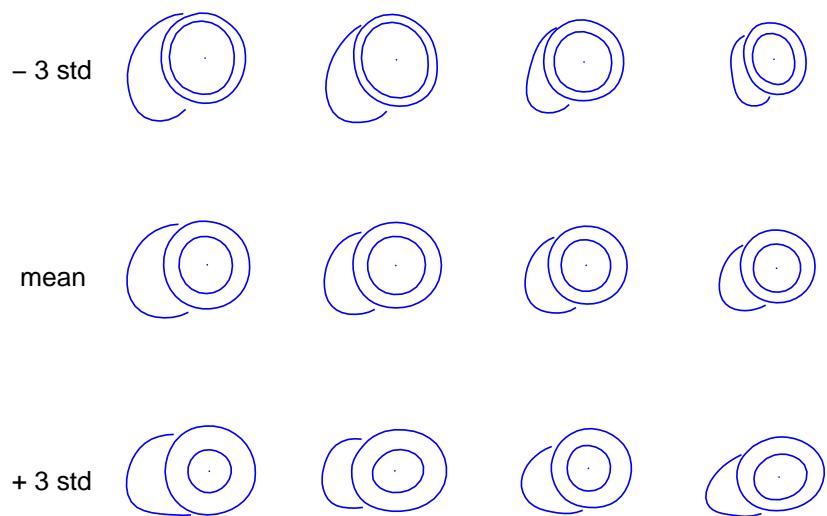
Figures 9.9–9.11 illustrate the properties of the joint shape model by showing principal shape modes 1–3 as deformations from the multi-slice mean shape. Figure 9.9 reveals that the first principal mode describes variations in orientation and size of the RV as well as the myocardium thickness for the first slice. The second principal mode in Figure 9.10 mostly describes variations in RV size and the third mode in Figure 9.11 accounts for variations in myocardium thickness and RV orientation. Judged from the three figures, the joint shape model seems very reasonable since most of the obvious variation in the RV and myocardium is explained.



**Figure 9.9:** First principal shape mode. From left to right: Slice 1–4.  
Top row: Mean shape  $-3$  std. dev. Middle row: Mean shape. Bottom row: Mean shape  $+3$  std. dev. Variations in orientation and size of the RV as well as myocardium thickness for the first slice are explained.

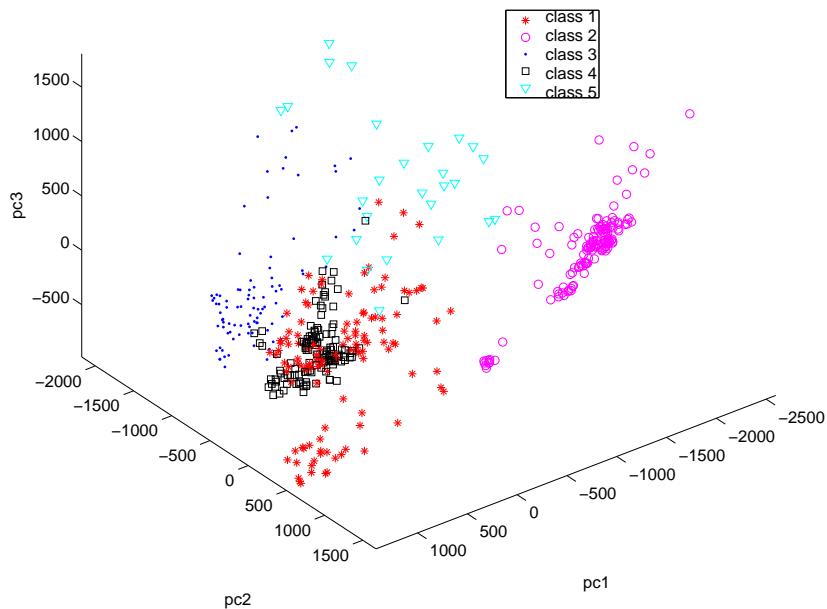


**Figure 9.10:** Second principal shape mode. From left to right: Slice 1–4.  
Top row: Mean shape  $-3$  std. dev. Middle row: Mean shape. Bottom row: Mean shape  $+3$  std. dev. Variations in size of RV are explained.



**Figure 9.11:** Third principal shape mode. From left to right: Slice 1–4.  
Top row: Meanshape  $-3$  std. dev. Middle row: Meanshape. Bottom row: Meanshape  $+3$  std. dev. Variations in myocardium thickness and RV orientation are explained.

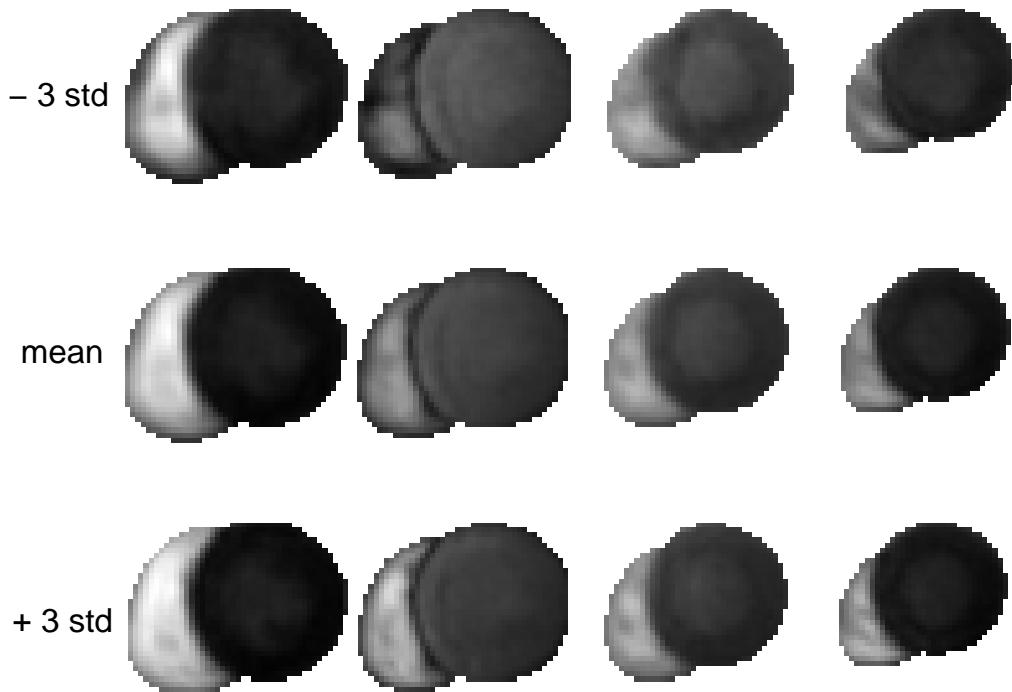
Now, the texture modelling is considered. The resulting texture classification of the data using five classes ( $n_c = 5$ ) is shown in Figure 9.12 in terms of the first three principal components. Judged from the figure, the clustering seems to be quite reasonable. However, the clustering is based on the full set of principal directions. Consequently, the plot is only an indicator of the clustering performance.



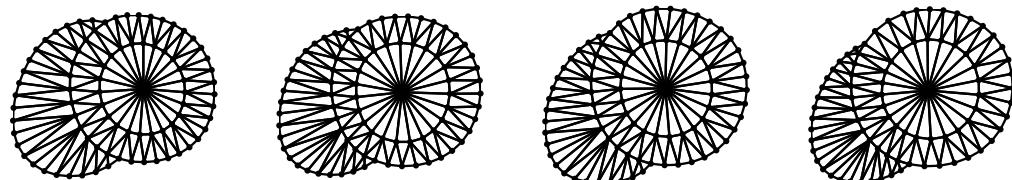
**Figure 9.12:** First, second and third principal component of 500 texture vectors from ten multi-slice perfusion sequences classified into five classes.

As described in Section 7.3 a model is built for each of the five classes. Figure 9.13 shows the first mode of texture variation for the model built from the training examples in the fifth class. The figure reveals that the first texture mode of class 5 mainly describes variations in the RV. The variation is hardly detectable, which is probably due to the fact that the clustering already accounts for a great deal of the variation. Class 5 is based on only 27 (out of 500) training examples, which inevitably have small texture variation.

As mentioned earlier, the pseudo-landmark in the centre of the LV was added to improve the Delaunay triangulation. This was extremely successful as the result displayed on the mean shape in Figure 9.14 indicates. The figure reveals that all triangles are close to being regular, which is the ideal case.



**Figure 9.13:** First principal texture mode. From left to right: Slice 1–4. Top row: Mean  $-3$  st. dev. Middle row: Mean. Bottom row: Mean  $+3$  st. dev. Variations in RV intensity are explained.



**Figure 9.14:** Delaunay triangulation of the meanshape. From left to right: Slice 1–4.

### 9.2.2 Cross validation of registration accuracy

The ideal way to evaluate the registration method would be to divide the data set into a training set and a test set. However, due to the small training set of only 10 patients, a leave-one-out cross validation was used. This infers that 10 training sets, excluding data from one patient at a time, were formed and registration was carried out on the data for the remaining (unseen) patient.

The stable period was set manually to the last 25 multi-slice frames of each sequence, i.e.  $S = 26$  except for patient 1. This patient had a delayed bolus passage and  $S = 36$  was used. This parameter could presumably easily be set auto-

matically using the extremum of the temporal image derivative. The model was automatically initialised in the  $P$ -th multi-slice frame using the approach from Section 7.7.

The initialisation strategy successfully located the heart in all patients, except patient number nine. Consequently, the CAAM for this patient was hand-initialised.

Two performance benchmarks were calculated for each model landmark:

- *Point to point (pt.pt.)*: Euclidean distance between corresponding landmarks of the model and the ground truth.
- *Point to curve (pt.crv.)*: Shortest distance to the ground truth curve.

These performance benchmarks are referred to as *landmark errors* in the following discussion. When grouped into mean values for a (multi-slice) shape these are referred to as *shape errors*.

Table 9.3 summarises the shape errors for all patients, while Table 9.4 expands on these numbers to show errors for each patient. Computation times for the initialisation and the image search are given in Table 9.5.

**Table 9.3:** Mean shape errors (shown in pixels).

	Mean	Std.dev.	Median	Min	Max
Pt.pt.	2.81	0.71	2.68	1.51	6.69
Pt.crv	1.25	0.36	1.17	0.66	3.79

**Table 9.4:** Shape errors for each patient (shown in pixels).

Patient	1	2	3	4	5	6	7	8	9	10
Pt.pt.										
Mean	2.50	2.81	3.45	3.23	2.68	2.53	2.62	3.26	2.75	2.26
Std.dev.	0.46	0.71	0.44	0.83	0.41	0.73	0.63	0.39	0.90	0.35
Median	2.45	2.74	3.45	2.99	2.62	2.40	2.52	3.28	2.59	2.21
Min	1.51	1.76	2.53	2.21	1.90	1.52	1.73	2.49	1.74	1.73
Max	3.50	4.56	4.39	5.13	3.72	5.34	5.14	4.06	6.69	3.72
Pt.crv.										
Mean	1.29	1.07	1.62	1.31	1.03	1.18	1.35	1.21	1.35	1.10
Std.dev.	0.23	0.15	0.35	0.22	0.21	0.48	0.39	0.16	0.54	0.24
Median	1.27	1.03	1.68	1.26	0.97	1.05	1.27	1.18	1.21	1.06
Min	0.78	0.85	0.86	1.03	0.75	0.66	0.98	0.94	0.80	0.83
Max	1.82	1.50	2.20	1.83	1.69	2.98	3.26	1.67	3.79	2.21

The statistics in Table 9.3 indicate a good overall registration accuracy. In Table 9.4 no patient sticks out with respect to shape errors and on average, good registration accuracy is obtained for all patients.

The computation times from Table 9.5 show that the dominating factor is the

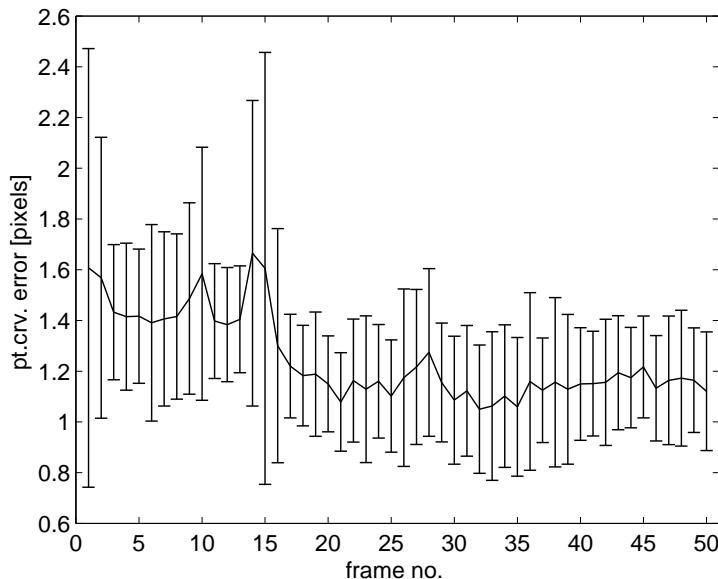
**Table 9.5:** Computation times (shown in seconds).

	Multi-slice sequence time	Mean frame time
Initialisation	34.0	0.7
CAAM search	9.9	0.2
Total	43.9	0.9

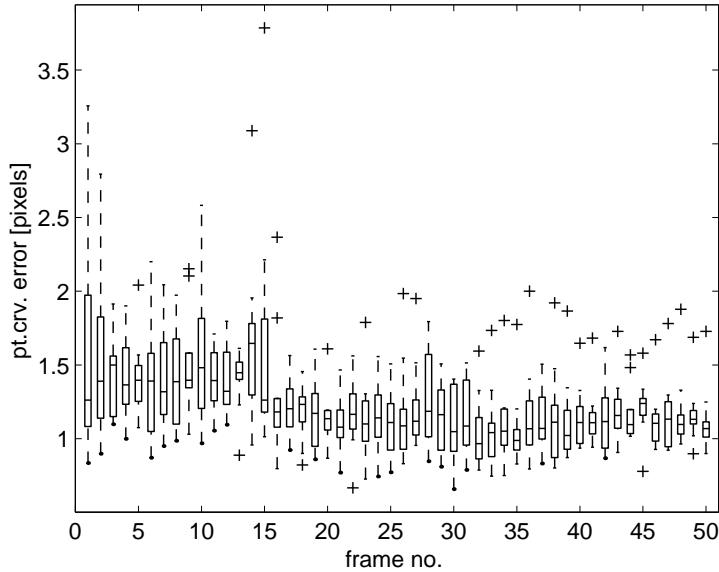
initialisation procedure which accounts for almost 80% of the computation time.

In Figure 9.15 – 9.17, shape and landmark errors will be illustrated by the pt.crv. measure. The pt.pt. measure is not shown since it gives very similar trends.

Shape errors versus frame number are depicted in Figure 9.15 and 9.16 as mean values over all patients with error bars (revealing trends and variation) and as box plots (detecting spread of data and outliers), respectively. Notice the large shape errors during the unstable period (frame 1–25) in both plots.

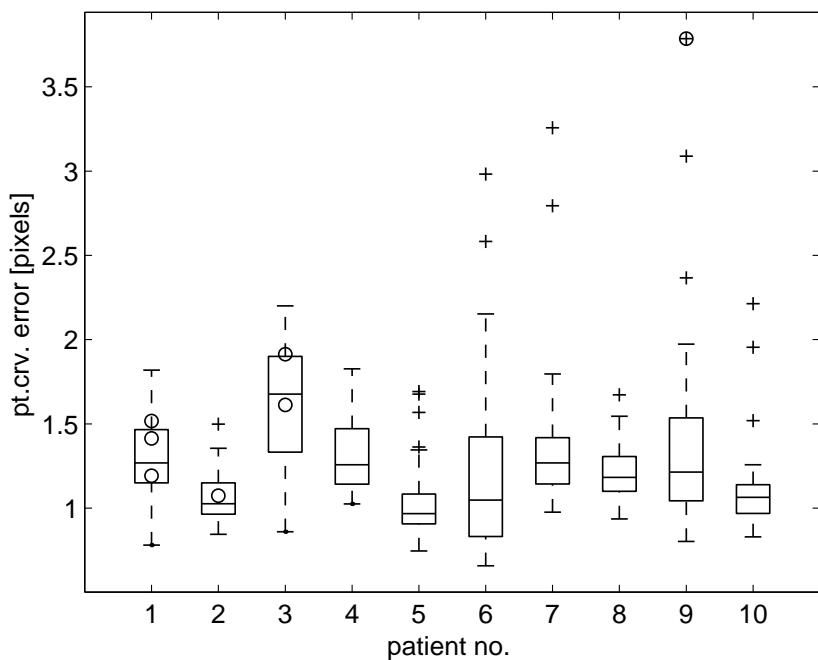


**Figure 9.15:** Pt.crv. shape errors for each multi-slice frame shown as mean over all patients. Error bars are one std. dev. Notice the large shape errors during unstable period (frame 1–25)



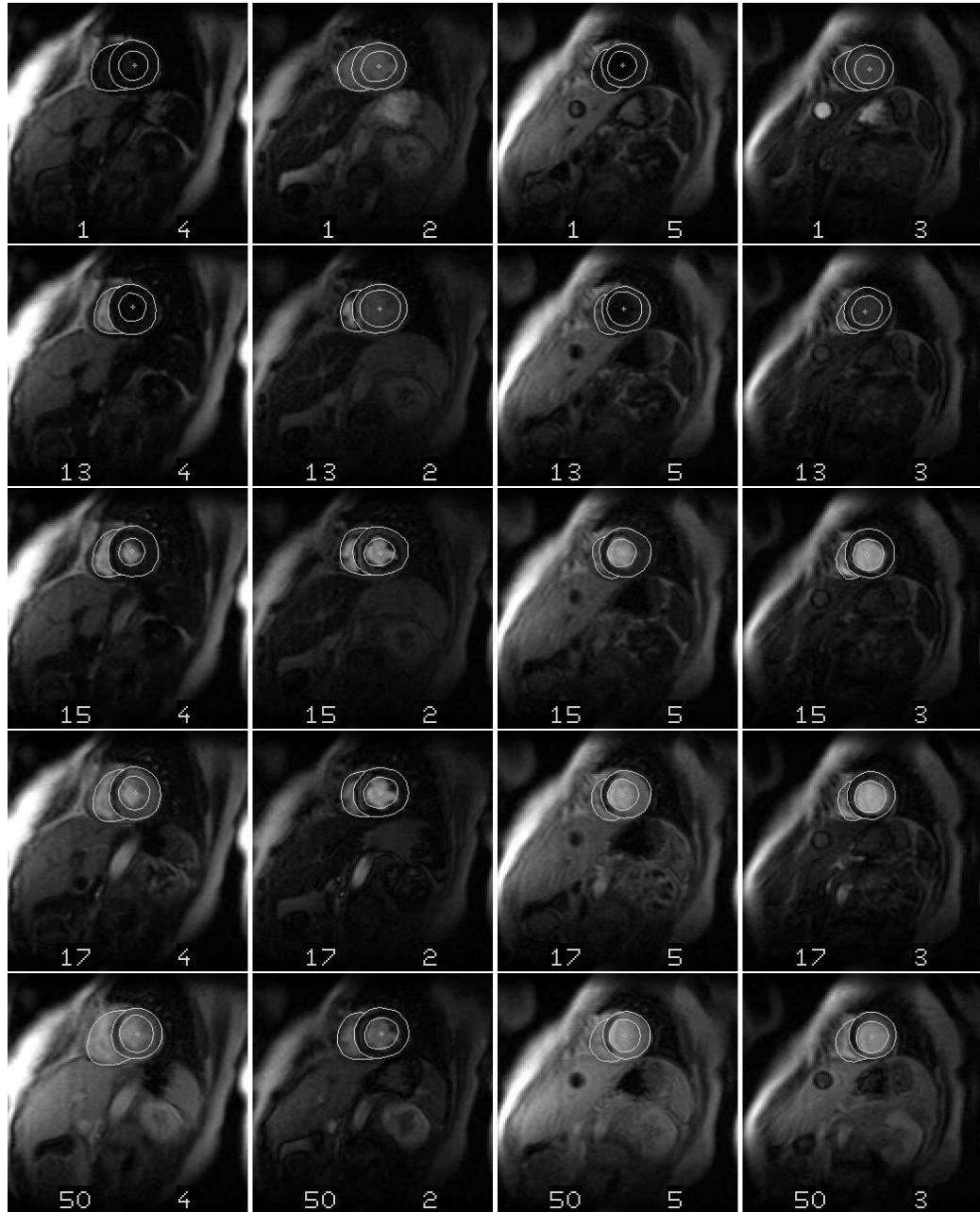
**Figure 9.16:** Box plots of the pt.crv. shape errors over all patients for each multi-slice frame ( $IQR=1.5$ ). Notice the large shape errors during unstable period (frame 1–25)

Plotting pt.crv. shape errors versus patient number in Figure 9.17 shows approximately 14 apparent search failures out of 500 multi-slice frames ( $14/500=2.8\%$ ). This number is obtained by adding the seven outliers (defined by using the top-most whisker of patient 3 as a reference) to the seven failures detected and flagged by Equation 7.13, indicated by circle in the figure. For patients 1–3 the replaced mean estimator was close to the ground truth shape, contrary to patient 9. From Figure 9.16 it is seen that all other failures occurred during the unstable period (since none of the errors in the stable period exceed 2 pixels.)

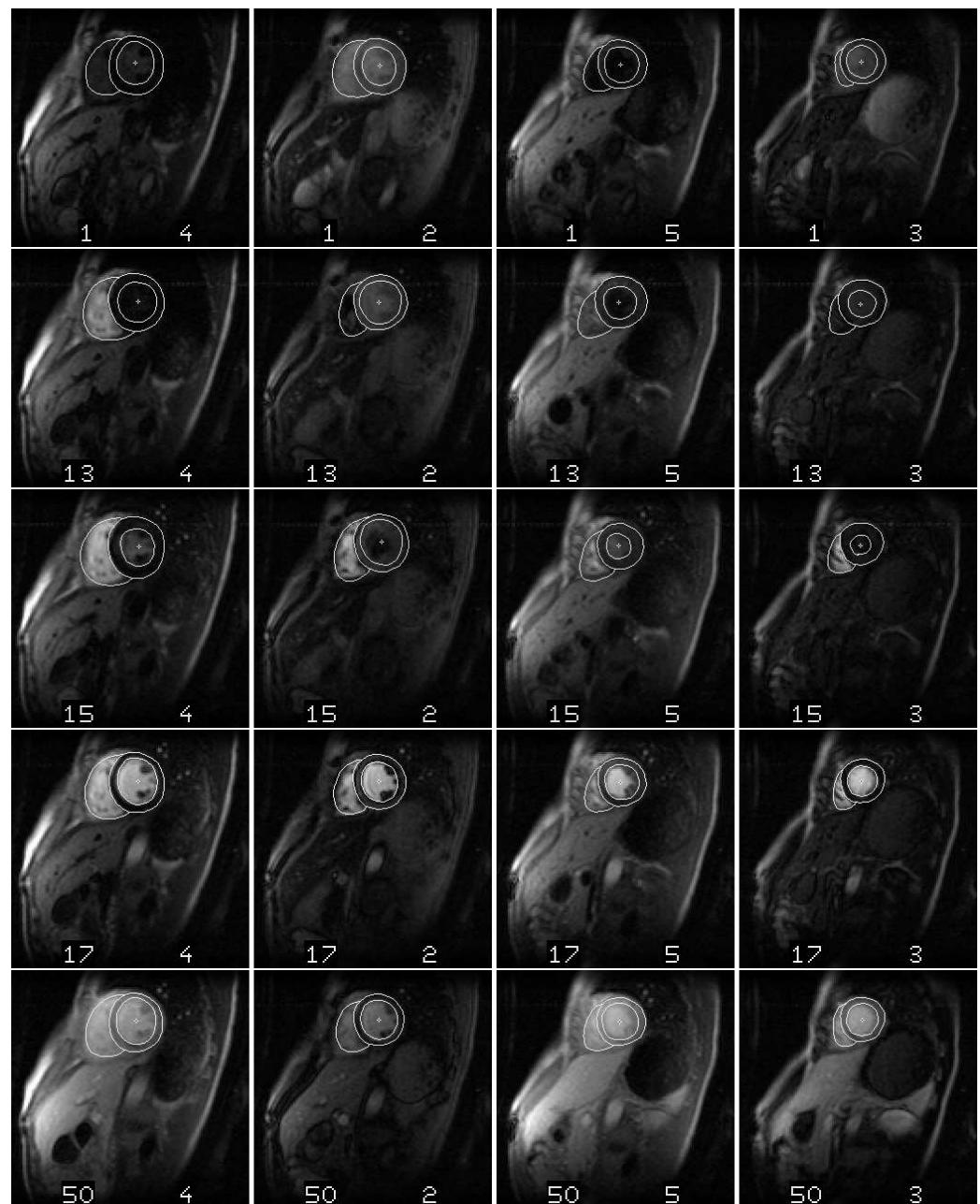


**Figure 9.17:** Box plots of the pt.crv. shape errors in all time frames for each patient (IQR=1.5). Pose prior enforcements are shown using circles. 14 apparent failures are detected including the ones denoted by circles.

To give a qualitative impression of the registration accuracy Figures 9.18 and 9.19 show registration results before, during and after the bolus passage for patient 8 and 10, respectively. Notice the dark region showing a severe perfusion deficit present in the anteroseptal and inferoseptal of patient 8 in Figure 9.18.

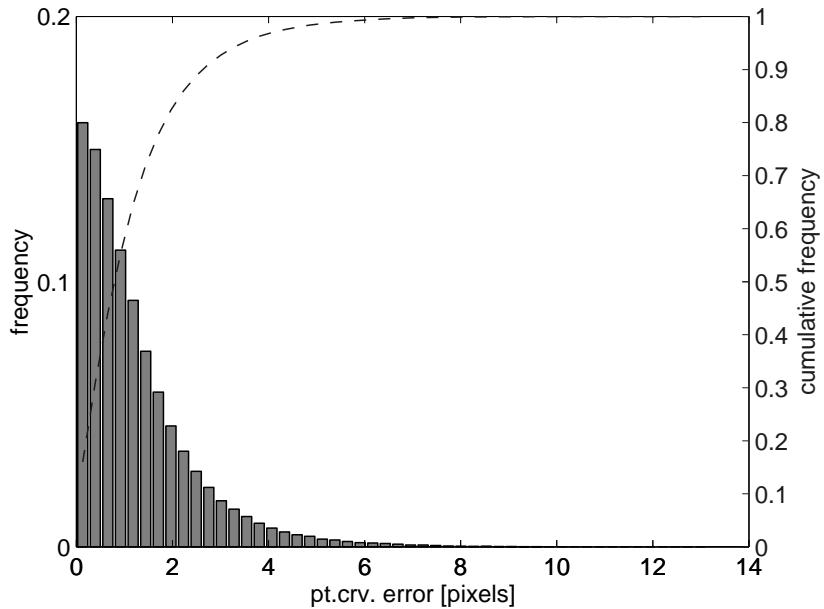


**Figure 9.18:** Registration results for patient 8 before (multi-slice frame 1), during (multi-slice frames 13, 15 and 17) and after (multi-slice frame 50) bolus passage. Notice the severe perfusion deficit present in the anteroapical and inferoseptal regions seen in the last two multi-slice frames (17 and 50).

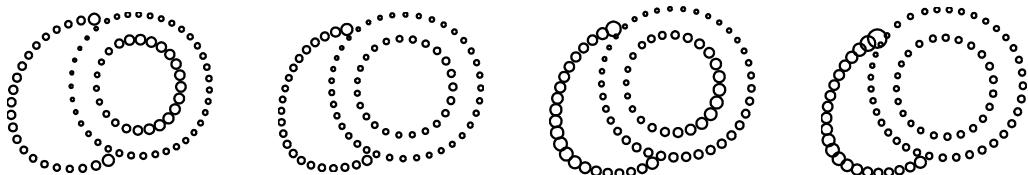


**Figure 9.19:** Registration results for patient 10 before (multi-slice frame 1), during (multi-slice frames 13,15 and 17) and after (multi-slice frame 50) bolus passage.

To complete the quantitative overview, Figure 9.20 shows the distribution of pt.crv. landmark errors in addition to their cumulative frequency. It is seen that approximately 80 percent of the landmark errors are below two pixels. Finally, Figure 9.21 illustrates the spatial distribution of the landmark errors by showing circles with radii proportional to the mean landmark error on the multi-slice mean shape. Errors on RV dominate together with errors on the LV endocardial contour in proximity of the papillary muscles, i.e. near the anterolateral and inferolateral wall, of the first and third slice.



**Figure 9.20:** Distribution of pt.crv. landmark errors for all multi-slice frames shown by bars. Cumulative frequency denoted by a broken line indicates that 80% of the errors are below two pixels.



**Figure 9.21:** Mean landmark errors plotted as circles on the multi-slice mean shape.

### 9.2.3 Discussion of registration results

The AAM based registration method produced acceptable results without manual interaction in nine out of ten cases, due to the described initialisation method. This

strategy failed for patient 9, for whom the heart was poorly centred in the slice field-of-view. Further, and more importantly, slices were shifted heavily towards the apex compared to the remaining nine patients, which obstructed a good fit of the LV-RV model in the initialisation process. Patient 9 is therefore considered an outlier and the observed limitations are accepted.

In general, less than three percent model fit failures were observed, judged from Figure 9.17. All occurred prior to, or during the bolus passage (unstable period). Half of these failures were not detected by Equation 7.13. This performance is considered convincing, although it indicates that registrations based on this limited training data should be reviewed by the operator, prior to making judgements upon automatically generated perfusion maps. The operator workload involved herein, corresponds approximately to performing manual annotation of a single multi-slice frame.

Figure 9.21 reveals that a substantial part of the landmark error stems from imperfections in the localisation of the right ventricle. LV registration errors are consequently smaller than the reported averages in the previous section. Although the inclusion of the RV adds specificity to the model, a hierarchical approach using a separate LV model initialised from the converged RV-LV model is straightforward and should be well worth to pursue.

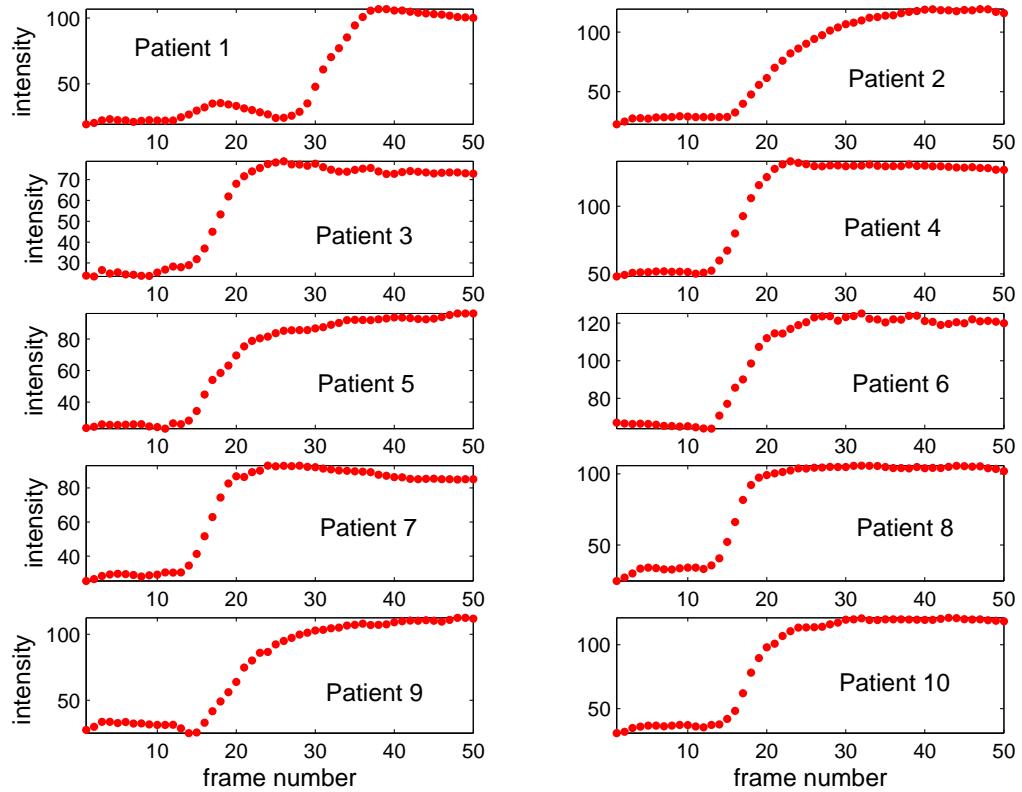
### 9.3 Perfusion assessment

Now that the registered perfusion sequences are provided, a semi-quantitative perfusion assessment can be carried out.

After extracting the myocardium from the registration results, smoothing over time was performed by averaging over adjacent frames. Additionally, spatial smoothing was performed by mean filtering inside the myocardium. Subsequently, SI-curves were generated for each pixel position.

To get an idea of the appearance of the SI-curves, Figure 9.22 shows the mean SI-curve for each patient. This mean curve is obtained by taking the mean intensity over the whole myocardium from slice 3 in each time frame.

The SI-curves seem quite reasonable even though the wash-out of the contrast agent is not reflected as in the theoretical SI-curve shown in Figure 8.2. This behaviour can make the determination of the peak position problematic. For patients 5 and 9 the wash-out does not appear at all. That is, the intensities are still increasing when the last time frame is reached. For the other patients, one can see a decrease (with good intension). The poor wash-out of the bolus is probably due to the fact that all the patients have acute myocardial infarction which means that more time is needed for the contrast to wash-out.



**Figure 9.22:** Mean SI-curve for all patients. Note the poor wash-out of contrast agent for patients 5 and 9 in particular.

The three perfusion parameters, maximum up-slope, peak and time-to-peak were calculated for each position in the myocardium and perfusion maps were generated for each of them. These are shown in Figures 9.23– 9.25 for all four slices of patients 3, 4 and 7. For the interested reader, refer to Appendix C.2 for perfusion maps of the remaining patients. The maps are interpreted by examining areas which appear darker than the surroundings. This indicates insufficient blood flow to the particular regions (ischemic regions). Note that the maps only indicate relative differences in blood flow and intensities can therefore neither be compared between slices nor patients. In the following discussion of the perfusion maps, the different regions of the myocardium will be referred to as labelled in Figure 8.3.

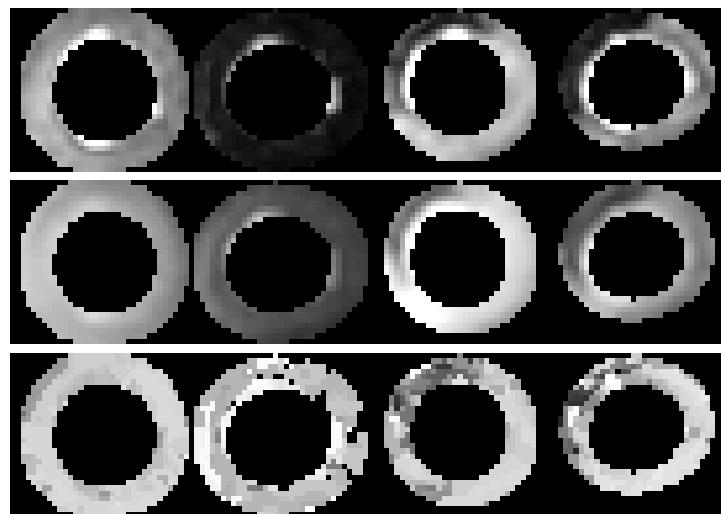
In Figure 9.23 ischemic regions are seen in the anteroseptal of slice 3 and 4 for all three parameters.

Figure 9.24 shows that patient 4 has insufficient blood flow in the anteroseptal. This is concluded from the perfusion maps of slice 3–4 for the maximum upslope, slice 4 for the peak and slice 2–4 for the time-to-peak.

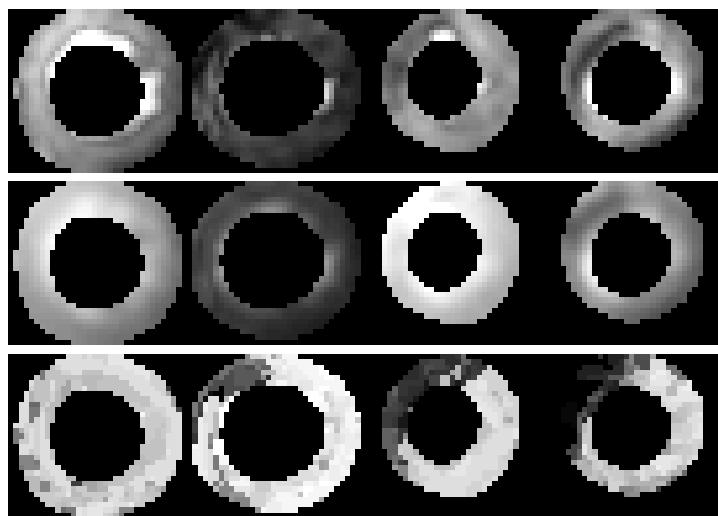
Figure 9.25 indicates that the ischemic regions for patient 7 appear at the inferior

and inferoseptal. This can be seen from the maps for slices 1,3 and 4 of the maximum upslope and peak and slices 3 and 4 for the time-to-peak parameter.

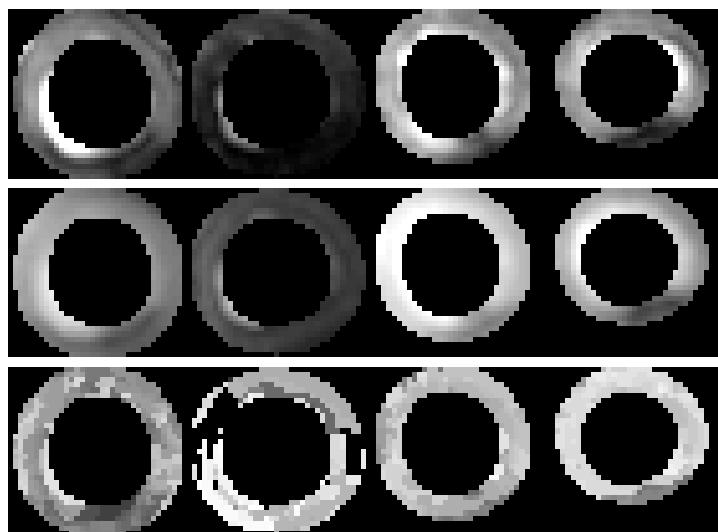
The bright intensities appearing at the endocardium border in some of the perfusion maps (for example slice 1 in maximum upslope map for all three patients) are probably due to minor registration failures, i.e. a small portion of the left ventricle has been included in the myocardium registration. This is concluded from the fact that the SI-curve for the LV is considerably steeper and with larger peak than the myocardium curve. This results in greater intensities in the perfusion maps for peak and maximum upslope.



**Figure 9.23:** Perfusion maps for patient 3 generated from automatic registration. From top to bottom: Maximum upslope, peak and time-to-peak. From left to right: Slice 1–4. Areas that appear darker than the surroundings indicate reduced blood flow (ischemic segments).



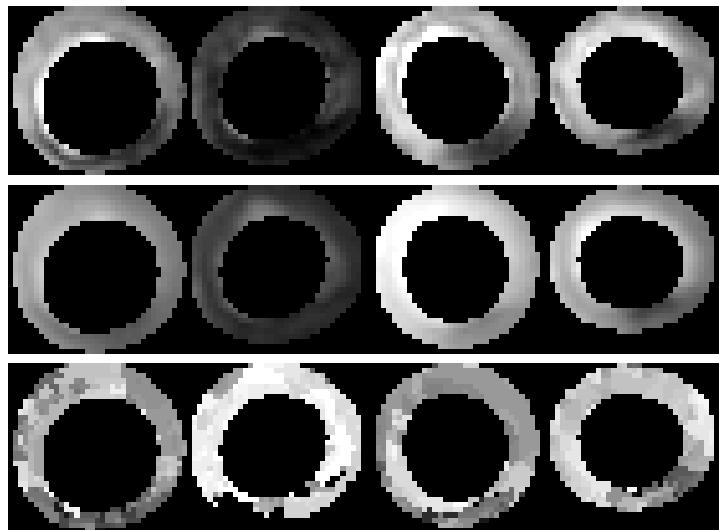
**Figure 9.24:** Perfusion maps for patient 4 generated from automatic registration. From top to bottom: Maximum upslope, peak and time-to-peak. From left to right: Slice 1–4. Areas that appear darker than the surroundings indicate reduced blood flow (ischemic segments).



**Figure 9.25:** Perfusion maps for patient 7 generated from automatic registration. From top to bottom: Maximum upslope, peak and time-to-peak. From left to right: Slice 1–4. Areas that appear darker than the surroundings indicate reduced blood flow (ischemic segments).

Perfusion maps were also generated for the ground-truth registered sequences. These are shown for patient 7 in Figure 9.26. These ground-truth maps indicate reduced blood flow in the inferior and inferoseptal regions. This is very similar to what was concluded from the perfusion maps obtained from the automatic

registration for same patient in Figure 9.25. In general, good consistency between the ground-truth and automatic perfusion maps was obtained and this may be verified by observing the remaining ground-truth perfusion maps in Appendix C.3.



**Figure 9.26:** Perfusion maps for patient 7 generated from ground-truth registration. From top to bottom: Maximum upslope, peak and time-to-peak. From left to right: Slice 1–4. Areas that appear darker than the surroundings indicate reduced blood flow (ischemic segments).

### Discussion of perfusion assessment

The SI-curves in Figure 9.22 indicated for most patients that wash-out of contrast agent had not occurred within the given time. As mentioned, this is probably because the patients all have myocardial infarction. In this relation, additional time frames in the sequences would be desirable. Furthermore, a comparison to healthy individuals would be interesting in order to confirm those thoughts.

The perfusion maps from the automatic registration method were proven to be able to detect ischemic areas of the myocardium in the three patients reported here. Similar results were obtained for most of the remaining patients and good consistency was obtained with respect to perfusion maps from the ground-truth registration.

The bright areas appearing at the endocardium in the perfusion maps indicate minor registration failures. These are quite difficult to avoid entirely and were also detected in the ground-truth maps, see for example slice 1 and 3 in maximum upslope map in Figure 9.26. As discussed in Chapter 3, Spreeuwiers et al. [61] have suggested sensitivity analysis on the perfusion parameters to overcome this

problem.

The analysis provided in this section forms a basis for the absolute quantification of blood flow. This is one of the future objectives of this study and in particular it will be interesting to see a comparison of blood-flow in the automatically registered perfusion sequences versus the ground-truth sequences.

## Chapter 10

# Summary and conclusions

### 10.1 Summary

A registration method for four-dimensional myocardial perfusion MR images has been presented. The method is based on Active Appearance Models, with two main extensions for the perfusion data. This includes modelling of the multi-slice sequences in addition to clustering of texture vectors in the temporal dimension.

To reduce manual work in generating a training set for the registration method, point correspondences were obtained in a semi-automatic manner, accomplished in two steps.

First, manual extraction of shape contours was carried out. In this purpose, a MATLAB tool was implemented. The tool enables the user to mark points arbitrarily on the outline and subsequently, the points are interpolated by a cubic spline to generate the dense shape contour.

Secondly, point correspondences on the extracted shape contours were obtained automatically by Minimum Description Length shape modelling. Before applying this method, the cardiac data needed a few adjustments. This included splitting the cardiac shape into five shape contours at anatomical and pseudo-landmarks. Due to impractical computation times, the method could not be applied to the full shape data. Consequently, optimisation was carried out on a non-temporal set and the optimal solution from this non-temporal set was applied to the remaining shapes in the set.

After automatically registering the data in a leave-one-out examination, a semi-quantitative perfusion assessment was carried out by providing perfusion maps for three perfusion parameters.

## 10.2 Conclusions

The objectives set forth in Chapter 1 were the following:

- To develop a tool for manual extraction of contours from myocardial perfusion data.
- To place landmarks automatically on the extracted contours by MDL shape modelling so that optimal point correspondence is obtained.
- To register the multi-slice image sequences using an AAM-based method.
- Perform semi-quantitative perfusion assessment on the registered image sequences.

These objectives have been achieved to a large extent as now discussed.

The contour extraction tool performed satisfactory for the manual marking of points on the shape outlines and gave smooth shape contours by applying cubic splines for interpolation. The annotation procedure naturally included the determination of anatomical landmarks. Consequently, a good shape model was already provided after sampling points equidistantly along the interpolated contours.

To further improve the model, point-correspondences were optimised by MDL shape modelling. The five non-temporal sub-contour optimisations resulted in 10–14% improvement in terms of description length and 10–28% improvement in terms of the total model variance. This is quite acceptable considering the fact that neither anatomical nor mathematical landmarks are present.

The crude approximation of assuming the parameterisation function ( $\psi$ ) to be constant throughout the slice sequence was applied to derive landmarks on the remaining shapes. This resulted in a reasonable shape model even though the overall improvement from the equidistant model was not considerable. However, it would have been desirable to avoid this approximation and optimise point correspondences on the full data set. The apparent obstacles for this to come true are the optimisation method (pattern search) which has poor computational complexity and the fact that the code is written in MATLAB. Even though the code is optimised with respect to MATLAB’s matrix and vector routines, it includes a considerable amount of for-loops which generally execute slowly in MATLAB.

From the above, it is concluded that the objective set forth on obtaining optimal point correspondences by MDL shape modelling was not achieved entirely even though the approximation resulted in an acceptable shape model.

By training the AAM on the near optimal training set from MDL, good registration results were produced. This is concluded from the registration accuracy of  $1.25 \pm 0.36$  pixels obtained in a leave-one-out cross validation. Since the largest landmark errors appeared at the RV, the average registration accuracy for the myocardium was even better than indicated by the overall average reported above.

Generally, the obtained registration accuracy is considered promising, especially since the data originates from patients diagnosed with acute myocardial infarction leading to image quality below average. This poor image quality was also observed at the training set formation stage, where it turned out that the manual extraction of the shape outlines could be very problematic.

Compared to other approaches mentioned in Chapter 3, the registration method is considered to have a noteworthy set of merits. These will be addressed in the next few paragraphs.

In general, the method has the advantage that it is based upon a well-understood and well-described framework; AAM, which also has a freely available implementation.

Some of the alternative approaches only correct for translational motion, disregarding rotation and deformable shape changes. By allowing complex deformations of the model, the AAM based registration method is not only suitable for motion correction due to imperfect ECG triggering but it could as well be generalised to multi-slice perfusion sequences without ECG triggering (resulting in a considerable shape variation across time). This type of scanning process is however not yet available.

To let the method generalise to new data, very few assumptions concerning the data content were made. Except for a few scalar parameters – which are dimensionless indices relating to the statistics of the actual data – all values are estimated from training data, rather than being explicitly coded into the method. This is considered a very fruitful approach, as the method easily adapts to new expert knowledge given by medical doctors.

Since only weak assumptions regarding sequence length and bolus passage position are present in the method, it is easily adapted to arbitrary-length perfusion sequences. The method only requires a coarse indication of the bolus passage position and a stable period of a reasonable number of time frames after the bolus, the latter being a standard requirement in perfusion MRI.

Contrary to the other approaches, the AAM based method provides simultaneous registration of all slices. Furthermore, the computation times obtained are well competitive to the other approaches, keeping in mind that they apply to modelling of four slices instead of one.

Due to the inherent representation of texture vectors in AAMs, a per-pixel correspondence over the complete perfusion sequence can be obtained by projecting the vectors to the shape-free reference frame. This enables the direct formation of SI-curves.

Regarding validation, both qualitative and quantitative validation of the AAM based registration method has been given. This includes comparison with manually defined (ground truth) shape boundaries. This validation methodology is a

clear advantage to most of the other approaches, which rely on relative comparison of parameters derived from the raw and the registered data. Consequently, these measurements are heavily dependent on the quality of the raw data.

The main advantages of the AAM based registration method are summarised below.

- It is based upon a well-understood and well-described framework; AAMs.
- Expert knowledge is provided in the learning-phase.
- Deformable modelling gives increased flexibility.
- It is easily adapted to arbitrary length perfusion sequences.
- All slices are modelled simultaneously.
- The registration provides per-pixel point-correspondence in a shape-free reference frame.
- Results are produced within a reasonable time frame.
- Good validation has been carried out, including quantitative comparison of automatic and ground-truth results.

The proposed registration method certainly has some limitations. The fact that the method is learning based limits it to synthesise only plausible instances as described by the training set. In some studies, papillary muscles, present on some patients but not others, are detected as a part of the myocardium. Obviously, this is not possible using the proposed method.

The semi-automatic formation of a training set was quite time consuming although the manual labour was reduced considerably. However, forming the training set is not considered a major disadvantage of the method. In fact, the need for ground-truth registration is always present if a sufficient validation is to be carried out.

Given the resulting training set, the data set processed in this study could not be registered fully automatically. However, manual interaction was minimal and corresponded to annotation of a single multi-slice frame.

Semi-quantitative perfusion assessment was carried out and provided perfusion maps for the three parameters: Maximum upslope, peak and time-to-peak. Ischemic regions were detected by the maps for most of the patients. Additionally, good consistency was obtained with respect to perfusion maps acquired from ground-truth registered data. Consequently, it is concluded that these results are promising for the ultimate goal of the examination, a fully quantitative perfusion assessment.

An invaluable property of MRI in general is that it is harmless for the patient unlike some other examination methods. Additionally, high spatial resolution of the images is a clear advantage. It is therefore desirable to utilise this technique to its full extent. The use of Myocardial perfusion MRI is steadily increasing but since it provides large numbers of images for each patient, (250 in this study), the examination must be computer aided. The largest step towards diminution of

manual labour in the examinations is to automate registration of the images.

Although automatic registration of perfusion MRI remains a challenge to medical image analysis, it is believed that the results presented in this thesis hold promise for the future.



## Chapter 11

# Future work

This chapter provides a few ideas for future work as new applications or as improvements of existing methods.

First of all, to be able to draw valid conclusions from the methods introduced, more data is needed for testing and validation.

The MDL implementation has given good results on small data sets but when the number of examples is increased as drastically as in this work, the computation time simply expands exponentially. To avoid the crude approximation of assuming no shape variation in the temporal dimension, a C/C++ code and/or another optimisation technique, such as a gradient based algorithm would be desirable.

The MDL shape modelling considered each shape extracted from one time-frame as one observation and thereby provided point correspondences across slices as well as patients. Given the multi-slice AAM based registration method, only point correspondences across patients are needed. This would most easily be achieved by optimising point correspondences across patients for one slice at a time.

Relating this to computational complexity, it was experienced that the computation times increased exponentially with increasing number of shapes. This indicates that an optimisation of the correspondences slice by slice (4 times 500 shapes for each sub-contour set) would not be as heavy as optimising the full set (2000 shapes at once for each sub-contour set). Consequently, this is the first step in avoiding the approximation discussed earlier.

Regarding the computation time for the AAM based registration method, the initialisation was a dominating factor as it accounted for almost 80% of the computation time. The initialisation method is a general approach based on parameter variations estimated from the training set. This method failed for one patient since it differed too much from the training set in terms of these parameters. Although general methods should be preferred, a possible alternative would be to use the method from [60], which is more specific for the perfusion data. This is based on

utilising the brightening effect of the contrast agent on the LV and RV. Initially, a smoothing in spatial and temporal dimensions is performed. Subsequently, a detection of the maximum intensity projection of the sequence followed by a local maximum detection determines the LV and RV centres. This is presumably a much faster approach than the general one used here.

Continuing with properties of the perfusion sequences, the determination of the bolus passage position could be automated. As mentioned earlier, this could be done by estimating the extremum of the temporal image derivative.

Further, regarding inspection of the perfusion sequences, the deformation property of the model is considered an advantage with respect to intra-patient shape variation. However, the shape variation across a slice-sequence is fairly small due to the ECG triggering. This a priori knowledge could be further utilised by putting less weight on shape deformations in the temporal dimension and thereby add more constraints to the search space.

Clustering of texture vectors was obtained by k-means classification. A more sophisticated method for modelling the texture variance is to use a mixture of Gaussian models. However, since, in practice the k-means clustering works quite well, this is not considered a high priority task.

Regarding the perfusion maps, in some cases high intensities around the endocardium indicated that a small part of the LV was included in the detected myocardium. As mentioned earlier, this could be improved by a sensitivity analysis of the perfusion parameters as introduced in [61].

The semi-quantitative perfusion assessment has room for improvements. This includes normalisation of the perfusion parameters with respect to the corresponding parameters for the LV. This would allow better comparison between slices and possibly patients. Additionally, this would presumably also enable overall color coding of the perfusion maps.

The ultimate future goal of this study is to provide a full quantification of blood flow from the automatic registration and compare to ground-truth registrations.

# Bibliography

- [1] N. Ablitt, J. Gao, P. Gatehouse, and G. Z. Yang. Motion decoupling and registration for 3D magnetic resonance myocardial perfusion imaging. In *International Conference Computational Science - ICCS 2002*, volume 2331, pages 285–294. Springer-Verlag, 2002.
- [2] P. T. Adeler. *Hemodynamic simulation of the heart using a 2D model and MR data*. PhD thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, 2001. URL [http://www.imm.dtu.dk/documents/ftp/phdliste/phd86\\\_01-a.html](http://www.imm.dtu.dk/documents/ftp/phdliste/phd86\_01-a.html).
- [3] All refer health. All refer health - pictures and images, 2003. URL <http://health.allrefer.com/pictures-images/>.
- [4] American Heart Association. International cardiovascular disease statistics, 2003. URL <http://www.americanheart.org/downloadable/heart/1059109452900FS06INT3REV7-03.pdf>.
- [5] R. Bansal and G. Funka-Lea. Integrated image registration for cardiac MR perfusion data. *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2002. 5th International Conference. Proceedings, Part I (Lecture Notes in Computer Science Vol.2488)*, pages 659–66, 2002.
- [6] F. Behloul, J. P. Roux, and R. Unterreiner. Automatic boundary detection of the left ventricle and measurement of local myocardial perfusion in MRI. *Computers in Cardiology 1997*, pages 145–148, 1997.
- [7] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape context. *IEEE Trans Pattern Analysis and Machine intelligence*, 24:509–522, 2002.
- [8] S. Benayoun, N. Ayache, and I. Cohen. Adaptive meshes and nonrigid motion computation. *Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on*, 1:730–732, 1994.
- [9] L. M. Bidaut and J. P. Vallee. Automated registration of dynamic MR images for the quantification of myocardial perfusion. *Jour. Magn. Reson. Imaging*, 13(4):648–655, 2001.
- [10] F. L. Bookstein. Landmark methods for forms without landmarks: morphometrics of group differences in outline shape. *Medical Image Analysis*, 1(3):

- 225–243, 1997. ISSN 13618415.
- [11] H. G. Bosch, S. C. Mitchell, B. P. F. Lelieveldt, F. Nijland, O. Kamp, M. Sonka, and J. H. C. Reiber. Active appearance-motion models for endocardial contour detection in time sequences of echocardiograms. *Proceedings of SPIE - The International Society for Optical Engineering*, 4322(1):257–268, 2001. ISSN 0277786x.
  - [12] J. G. Bosch, S. C. Mitchell, B. P. F. Lelieveldt, F. Nijland, O. Kamp, M. Sonka, and J. H. C. Reiber. Fully automated endocardial contour detection in time sequences of echocardiograms by active appearance motion models. *Computers in Cardiology*, pages 93–96, 2001. ISSN 02766574.
  - [13] J. G. Bosch, S. C. Mitchell, B. P. F. Lelieveldt, F. Nijland, O. Kamp, M. Sonka, and J. H. C. Reiber. Automatic segmentation of echocardiographic sequences by active appearance motion models. *Medical Imaging, IEEE Transactions on*, 21(11):1374–1383, 2002. ISSN 02780062.
  - [14] L. Bracoud, F. Vincent, C. Pachai, E. Canet, P. Croisille, and D. Revel. Automatic registration of MR first-pass myocardial perfusion images. In *Functional Imaging and Modeling of the Heart, FIMH 2003*, volume 2674 of *Lecture Notes in Computer Science*, pages 215 – 223, Lyon, France, 2003. Springer Verlag.
  - [15] M. Breeuwer, M. Quist, L. Spreeuwiers, I. Paetsch, N. Al-Saadi, and E. Nagel. Towards automatic quantitative analysis of cardiac MR perfusion images. *International Congress Series*, 1230:967–973, 2001. ISSN 05315131.
  - [16] M. Breeuwer, L. Spreeuwiers, and M. Quist. Automatic quantitative analysis of cardiac MR perfusion images. *Proceedings of the SPIE - The International Society for Optical Engineering*, 4322(1-3):733–42, 2001. ISSN 0277786x.
  - [17] K. Conradsen. *En introduktion til statistik - bind 2, 6.udgave*. IMM, DTU, 2002.
  - [18] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In *Proc. European Conf. on Computer Vision*, volume 2, pages 484–498. Springer, 1998.
  - [19] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *IEEE Trans. on Pattern Recognition and Machine Intelligence*, 23(6):681–685, 2001.
  - [20] T. F. Cootes and C. J. Taylor. *Statistical Models of Appearance for Computer Vision*. Tech. report, University of Manchester, 2001. URL <http://www.isbe.man.ac.uk/~bim/>.
  - [21] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models – their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.
  - [22] T. F. Cootes, G. V. Wheeler, K. N. Walker, and C. J. Taylor. Coupled-view active appearance models. *BMV2000. Proceedings of the 11th British Machine Vision Conference*, pages 52–61 vol.1, 2000.
  - [23] R. H. Davies. *Learning Shape: Optimal Models for Analysing Natural Variability*. PhD thesis, Department of Imaging Science and Biomedical Engineering, 2001.

- ing, University of Manchester, Stopford building, University of Manchester, 2002.
- [24] R. H. Davies, T. F. Cootes, and C. J. Taylor. A minimum description length approach to statistical shape modelling. *Information Processing in Medical Imaging. 17th International Conference, IPMI 2001. Proceedings (Lecture Notes in Computer Science Vol.2082)*, pages 50–63, 2001.
  - [25] R. H. Davies, C. J. Twining, T. F. Cootes, J. C. Waterton, and C. J. Taylor. A minimum description length approach to statistical shape modeling. *Medical Imaging, IEEE Transactions on*, 21(5):525–537, 2002. ISSN 02780062.
  - [26] C. Dornier, M. K. Ivancevic, P. Thevenaz, and J-P. Vallee. Improvement in the quantification of myocardial perfusion using an automatic spline-based registration algorithm. *Journal of Magnetic Resonance Imaging*, 18(2):160–168, 2003. ISSN 10531807.
  - [27] I. L. Dryden and K. V. Mardia. *Statistical Shape Analysis*. John Wiley & Sons, 1998.
  - [28] A. Dubb, B. Avants, R. Gur, and J. Gee. Shape characterization of the corpus callosum in schizophrenia using template deformation. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI*, volume 2, pages 381–388, 2002.
  - [29] G. J. Edwards, C. J. Taylor, and T. F. Cootes. Interpreting face images using active appearance models. In *Proc. 3rd IEEE Int. Conf. on Automatic Face and Gesture Recognition*, pages 300–5. IEEE Comput. Soc, 1998.
  - [30] A. Ericsson and K. Åström. Minimizing the description length using steepest descent. In *BMVC2003. Proceedings of the 14th British Machine Vision Conference, Norwich UK*, volume 2, pages 251–260, September 2003.
  - [31] M. Fleute and S. Lavallee. Building a complete surface model from sparse data using statistical shape models: application to computer assisted knee surgery. *Medical Image Computing and Computer-Assisted Intervention - MICCAI'98. First International Conference. Proceedings*, pages 879–87, 1998.
  - [32] E. Forgey. Cluster analysis of multivariate data. *Biometrics*, 21:768, 1965.
  - [33] A.F. Frangi, D. Rueckert, J.A. Schnabel, and W.J. Niessen. Automatic 3D ASM construction via atlas-based landmarking and volumetric elastic registration. *Information Processing in Medical Imaging. 17th International Conference, IPMI 2001. Proceedings (Lecture Notes in Computer Science Vol.2082)*, pages 78–91, 2001.
  - [34] C. M. Gallippi and E. T. Gregg. Automatic image registration for MR and ultrasound cardiac images. *Information Processing in Medical Imaging. 17th International Conference, IPMI 2001. Proceedings (Lecture Notes in Computer Science Vol.2082)*, pages 148–54, 2001.
  - [35] J. Gao, N. Ablitt, A. Elkington, and G. Z. Yang. Deformation modelling based on PLSR for cardiac magnetic resonance perfusion imaging. *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2002. 5th International Conference. Proceedings, Part I (Lecture Notes in Computer Science Vol.2488)*, pages 612–19, 2002.

- [36] S. N. Gupta, M. Solaiyappan, G. M. Beach, A. E. Arai, and T. K. F. Foo. Fast method for correcting image misregistration due to organ motion in time-series MRI data. *Magnetic Resonance in Medicine*, 49(3):506–14, 2003. ISSN 07403194.
- [37] A. Hill and C. J. Taylor. Automatic landmark generation for point distribution models. *BMVC94. Proceedings of the 5th British Machine Vision Conference*, pages 429–38 vol.2, 1994.
- [38] J. P. Hornak. The basics of MRI – an online textbook, 2004. URL <http://www.cis.rit.edu/htbooks/mri/>.
- [39] L. Hsu, K. L. Rhoads, A. H. Aletras, and A. E. Arai1. Surface coil intensity correction and non-linear intensity normalization improve pixel-resolution parametric maps of myocardial MRI perfusion. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003, 6th Int. Conference, Montréal, Canada, LNCS 2879*, pages 975–976. Springer, 2003.
- [40] M. Jerosch-Herold, C. Swingen, and R. T. Seethamraju. Myocardial blood flow quantification with MRI by model-independent deconvolution. *Medical Physics*, 29(5):886–97, 2002. ISSN 00942405.
- [41] C. Kambhamettu and D. B. Goldgof. Point correspondence recovery in non-rigid motion. *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR '92., 1992 IEEE Computer Society Conference on*, pages 222–227, 1992.
- [42] A.C.W. Kotcheff and C. J. Taylor. Automatic construction of eigenshape models by direct optimization. *Medical Image Analysis*, 2(4):303–314, 1998. ISSN 13618415.
- [43] H. B. W. Larsson, T. Fritz-Hansen, E. Rostrup, L. Søndergaard, P. Ring, and O. Henriksen. Myocardial perfusion modeling using MRI. *Magnetic Resonance in Medicine*, 35:716–726, 1996.
- [44] B. P. F. Lelieveldt, S. C. Mitchell, R. J. van der Geest, H. G. Bosch, M. Sonka, and J. H. C. Reiber. Time continuous segmentation of cardiac MR images using active appearance motion models. In *Computer Assisted Radiology and Surgery, CARS 2001, International Congress Series*, volume 1230, pages 961–966. Elsevier Science, 2001.
- [45] B. P. F. Lelieveldt, M. Uzumcu, R. J. van der Geest, J. H. C. Reiber, and M. Sonka. Multi-view active appearance models for consistent segmentation of multiple standard views: application to long- and short-axis cardiac MR images. *International Congress Series*, 1256:1141–1146, 2003. ISSN 05315131.
- [46] A. Lundervold, N. Duta, T. Taxt, and A. K. Jain. Model-guided segmentation of corpus callosum in MR images. In *Computer Vision and Pattern Recognition. IEEE Comput. Soc*, 1999. ISBN 0769501494.
- [47] T. Machnig, N. Wilke, G. Engels, K. Bachmann, F. C. Simm, J. Ellermann, J. Zhang, X. Ya, H. Merkel, G. Path, R. J. Bache, and K. Ugurbil. MR measurements of gd-dtpa kinetics for estimation of myocardial perfusion. *Computers in Cardiology 1991, Proceedings.*, pages 5–8, 1991.
- [48] S. Mitchell, B. Lelieveldt, R. Geest, H. Bosch, J. Reiber, and M. Sonka.

- Time continuous segmentation of cardiac MR image sequences using active appearance motion models. In *Medical Imaging 2001: Image Processing, San Diego CA, SPIE*, volume 1, pages 249–256. SPIE, 2001.
- [49] O. Muhling, M. Jerosch-Herold, M. Nabauer, and N. Wilke. Assessment of ischemic heart disease using magnetic resonance first-pass perfusion imaging. *Herz*, 28(2):82–89, 2003. ISSN 03409937.
  - [50] E. Nagel, N. Al-Saadiand, and E. Fleck. Cardiovascular magnetic resonance: Myocardial perfusion. *Herz*, 25(4):409–416, 2000. ISSN 03409937.
  - [51] H. B. Nielsen. Cubic splines, oct 1998. URL <http://www.imm.dtu.dk/pubdb/p.php?655>.
  - [52] E. Oost, B. P. F. Lelieveldt, G. Koning, M. Sonka, and J. H. C. Reiber. Left ventricle contour detection in X-ray angiograms using multi-view active appearance models. *Proceedings of SPIE - The International Society for Optical Engineering*, 5032 I:394–404, 2003. ISSN 0277786X.
  - [53] J. R. Panting, P. D. Gatehouse, G. Z. Yang, M. Jerosch-Herold, N. Wilke, D. N. Firmin, and D. J. Pennell. Original research - echo-planar magnetic resonance myocardial perfusion imaging: Parametric map analysis and comparison with thallium spect. *Journal of Magnetic Resonance Imaging*, 13(2):192–200, 2001. ISSN 10531807.
  - [54] A. Rangarajan, H. Chui, E. Mjolsness, S. Pappu, L. Davachi, P. Goldman-Rakic, and J. Duncan. A robust point-matching algorithm for autoradiograph alignment. *Medical Image Analysis*, 1:379–398, 1997.
  - [55] D. Rueckert, L.I. Sonoda, C. Hayes, D.L.G. Hill, M.O. Leach, and D.J. Hawkes. Nonrigid registration using free-form deformations: application to breast MR images. *Medical Imaging, IEEE Transactions on*, 18(8):712–721, 1999. ISSN 02780062.
  - [56] T. B. Sebastian, P. N. Klein, and B. B. Kimia. On aligning curves. *IEEE Trans Pattern Analysis and Machine intelligence*, 25:116–124, 2003.
  - [57] C. E. Shannon. A mathematical theory of communication. *Bell Systems Technical Journal*, 27:379–423 and 623–656, 1948.
  - [58] D. A. Skoog, F. J. Holler, and T. A. Nieman, editors. *Principles og Instrumental Analysis*. Harcourt Brace College Publishers, 1998, fifth edidtion.
  - [59] M. Sonka, B. P. F. Lelieveldt, S. C. Mitchell, J. G. Bosch, R. J. van der Geest, and J. H. C. Reiber. Active appearance motion model segmentation. *Proceedings Second International Workshop on Digital and Computational Video*, pages 64–68, 2001.
  - [60] L. Spreeuwers and M. Breeuwer. Automatic detection of the myocardial boundaries of the right and left ventricle in MR cardio perfusion scans. *Proceedings of SPIE - The International Society for Optical Engineering*, 4322 (3):1207–1217, 2001. ISSN 0277786X.
  - [61] L. J. Spreeuwers, F. Wierda, and M. Breeuwer. Optimal myocardial boundary estimation for MR cardio perfusion measurements using sensitivity analysis. *Computers in Cardiology, 2002*, pages 197–200, 2002. ISSN 02766547.
  - [62] M. B. Stegmann. Object tracking using active appearance models. In *Proc.*

- 10th Danish Conference on Pattern Recognition and Image Analysis, Copenhagen, Denmark*, volume 1, pages 54–60. DIKU, 2001.
- [63] M. B. Stegmann. 4D cardiac viewer, may 2003. URL <http://www.imm.dtu.dk/pubdb/p.php?2466>.
  - [64] M. B. Stegmann, R. H. Davies, and C. Ryberg. Corpus callosum analysis using MDL-based sequential models of shape and appearance. In *International Symposium on Medical Imaging 2004, San Diego CA, SPIE*. SPIE, feb 2004 (in press).
  - [65] M. B. Stegmann, B. K. Ersbøll, and R. Larsen. FAME – a flexible appearance modelling environment. *IEEE Trans. on Medical Imaging*, 22(10):1319–1331, 2003.
  - [66] H. D. Tagare. Shape-based nonrigid correspondence with application to heart motion analysis. *Medical Imaging, IEEE Transactions on*, 18(7):570–579, 1999. ISSN 02780062.
  - [67] The Economist. Medical imaging – MRI’s inside story. *The Economist Technology Quarterly*, 369(8353):24–26, Dec. 4th 2003.
  - [68] H. H. Thodberg. Minimum description length shape and appearance models. In *Information Processing in Medical Imaging, IPMI*, page 12. Springer, 2003.
  - [69] H. H. Thodberg and H. Olafsdottir. Adding curvature to minimum description length shape models. In *BMVC2003. Proceedings of the 14th British Machine Vision Conference*, volume 2, pages 251–260, 2003.
  - [70] G. Z. Yang, P. D. Gatehouse, J. Panting, P. Burger, D. J. Pennell, and D. N. Firmin. Motion analysis for magnetic resonance myocardial perfusion imaging. *Image Processing and Its Applications, 1997., Sixth International Conference on*, 2:838–842 vol.2, 1997.

## Appendix A

### Data file structure

The raw data from the MR scanner was received on the "simple file format" (*sf*). The format consists of a header file (*.sfh*) and a data file (*.sfd*) for each image frame of the perfusion sequence. The header file consists of basic fields, which define the properties of the data, for example data format, resolution etc. The data file holds the data values on the format specified by the header file. More information on the *sf* format can be found at <http://www.drcmr.dk/software/>.

In order to visualise the data and reduce the amount of files, it was converted to the time volume format (*.tvi*) which holds the four-dimensional data for each patient in one file. The *.tvi*-files can be viewed in the 4D viewer [63] frame by frame and slice by slice.

For multi slice modelling, the data was also stored on the *.hif* format, where each file includes one multi-slice time frame where the slices are concatenated vertically. None of the file conversions loses data information.



## Appendix B

# Annotation tool – instructions

### B.1 Help file for annotation tool

This script finds all tvi-files and executes the function drawContours to annotate each frame in each one of them. A spline is fitted to each user annotated contour.

Annotation should be performed in the following way, using:  
 left mouse button for marking points  
 right mouse button to mark last point on the contour

Contrast: Press:  
 + to increase contrast  
 - to decrease contrast

Brightness: Press:  
 Right-arrow to increase brightness  
 Left-arrow to decrease brightness

1. Annotate Epicardial (Outer contour of myocardial)  
 First point should be placed at the upper anatomical landmark where the RV and epicardial meet. Annotate clockwise.

2. Annotate Endocardial (Inner contour of myocardial)  
 Annotate clockwise.

3. Annotate RV using 12-14 points. First point should be set at the previously marked landmark and last point should be set at the lower anatomical landmark where the epicardial and RV meet.

While annotating, the user can press  
 'u' for undoing last point  
 'a' for undoing all points currently marked on the contour in question.

After annotating a contour, the spline is plotted. The user can press: 'ENTER' or start annotating the next contour to accept the annotation 'r' to repeat annotation on the same contour.

The script saves the stretchbox and n.o. evaluation points for the spline in the workspace workspName1. Annotations, landmarks and a counter are saved in the workspace workspName2. If the run is stopped for some reasons and annotations proceeded later, the script checks the counter in the saved workspace, determines last annotation and proceeds where it stopped last time.

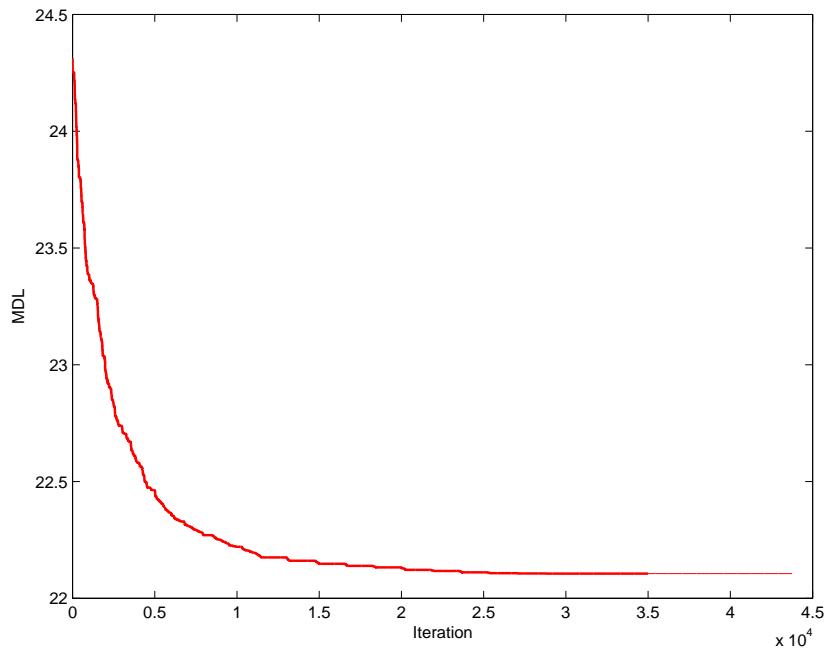
NB!! All tvi-files must have same number of slices and frames

## Appendix C

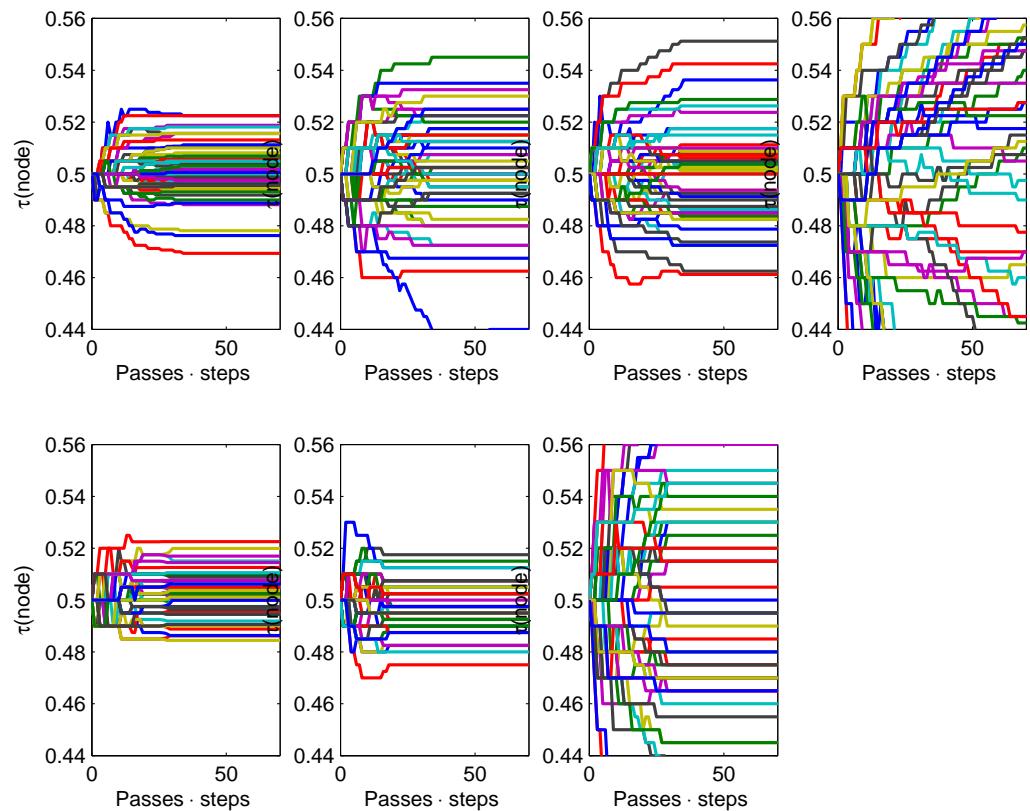
# Additional results

### C.1 MDL results, non-temporal set

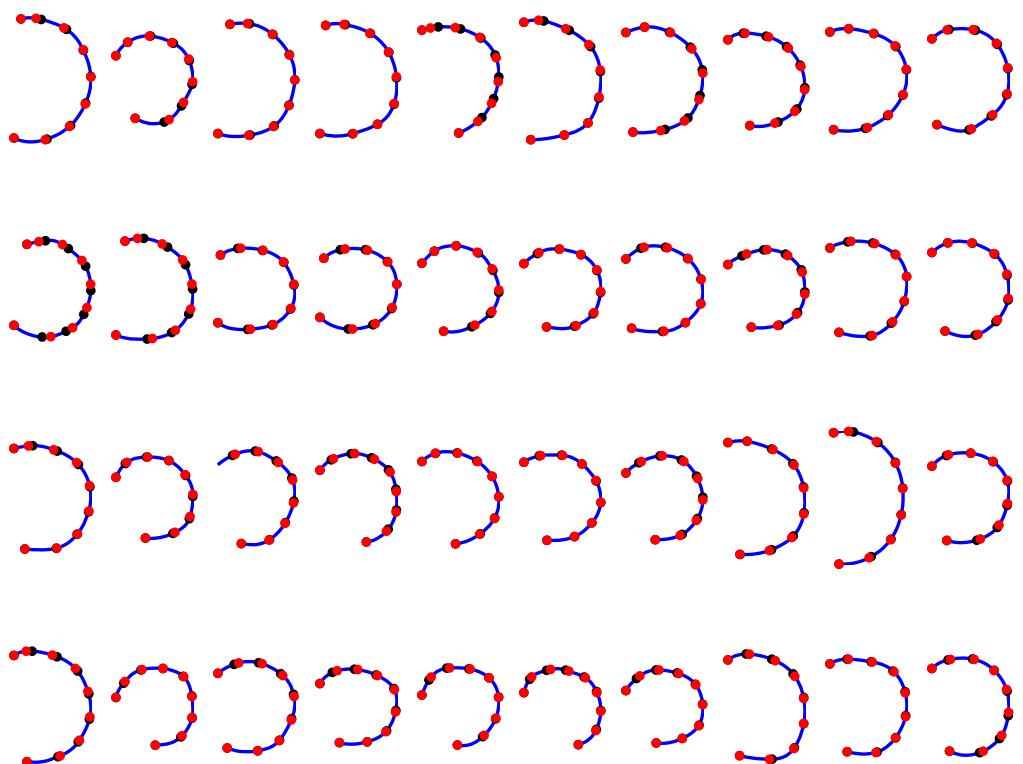
#### C.1.1 MDL Results for right epicardial contour set



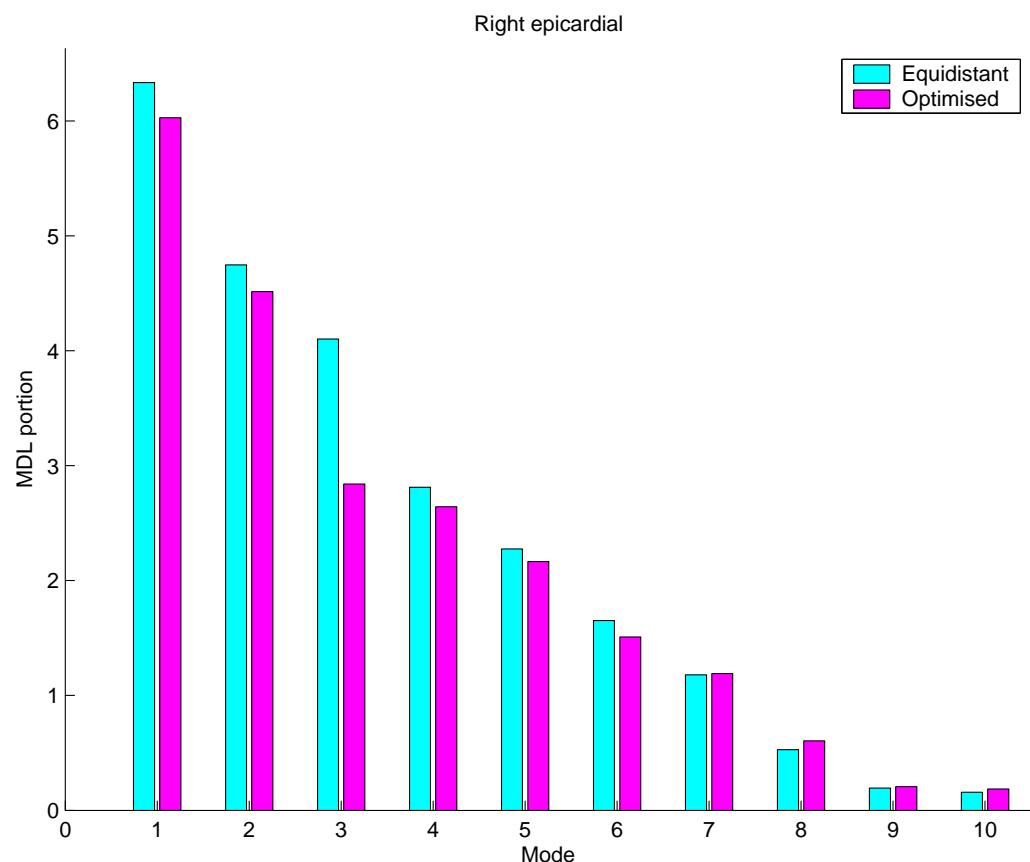
**Figure C.1:** Convergence of MDL optimisation for the right epicardial contour set. Note that the iteration number accounts for #passes · #nodes · #steps · #examples.



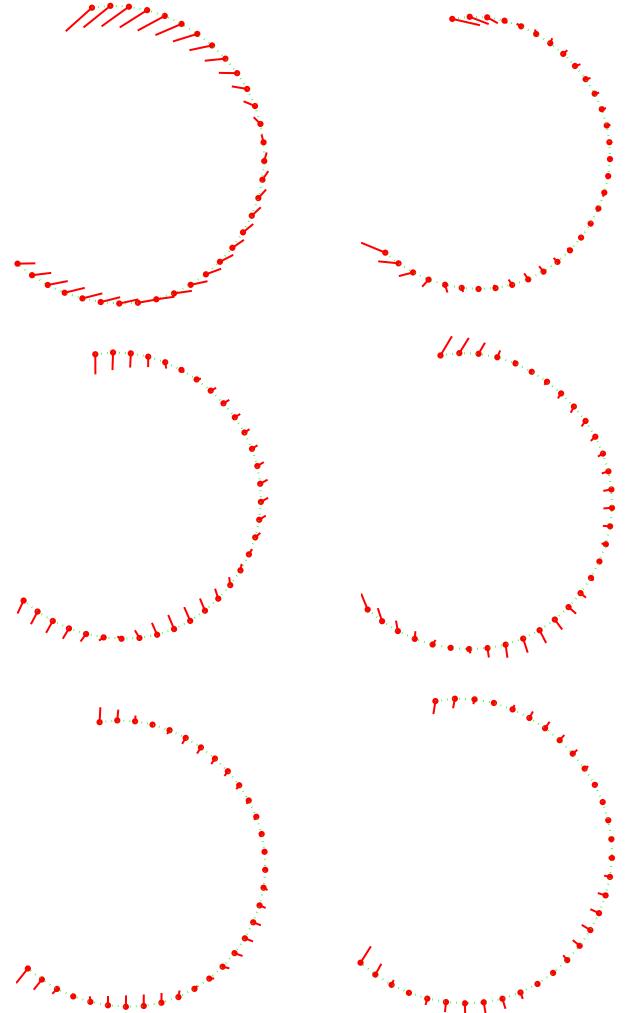
**Figure C.2:** Development of the seven node positions for optimisation of right endocardial contour set. Each node plot shows the value of  $\tau(\text{node})$  plotted versus  $\# \text{passes} \cdot \# \text{steps}$  for all shapes in the set.



**Figure C.3:** Fixed points and node positions for the right epicardial contour set. Equidistant (black) and MDL optimised (red) solution. Each column holds slice 1–4 for each patient.

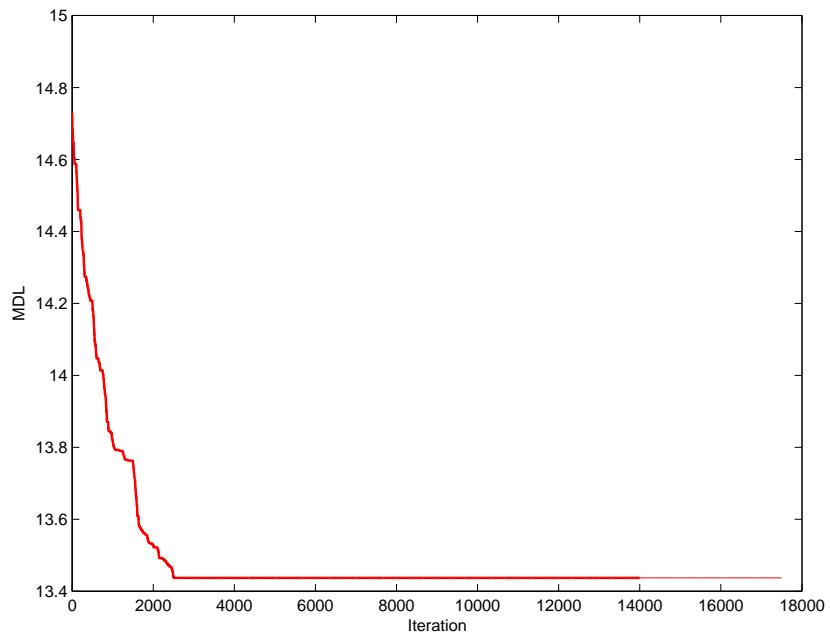


**Figure C.4:** Contribution of the first 10 eigenvalues to the objective function. Comparison of equidistant landmarks and MDL optimised landmarks for the right epicardial set.

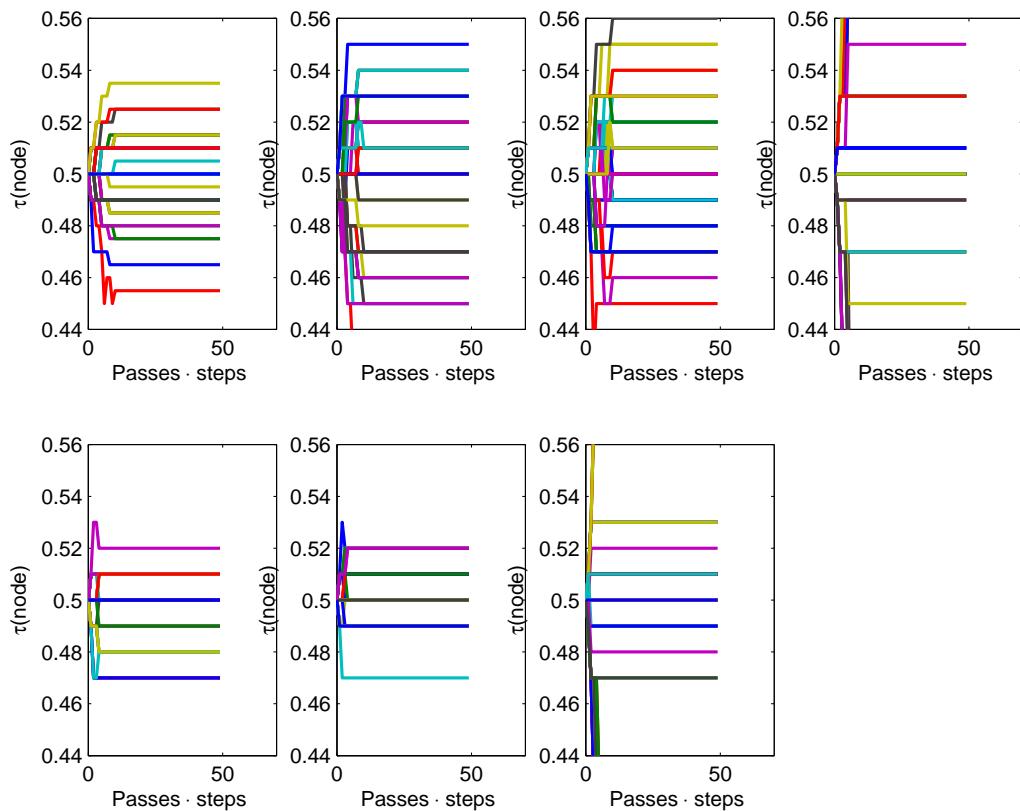


**Figure C.5:** Comparison of principal mode 3,4 and 5 (from top to bottom) before and after optimisation of the right epicardial contour set. Left: Equidistant landmarks, right: MDL optimised landmarks. The whiskers denote 5 standard deviations from mean.

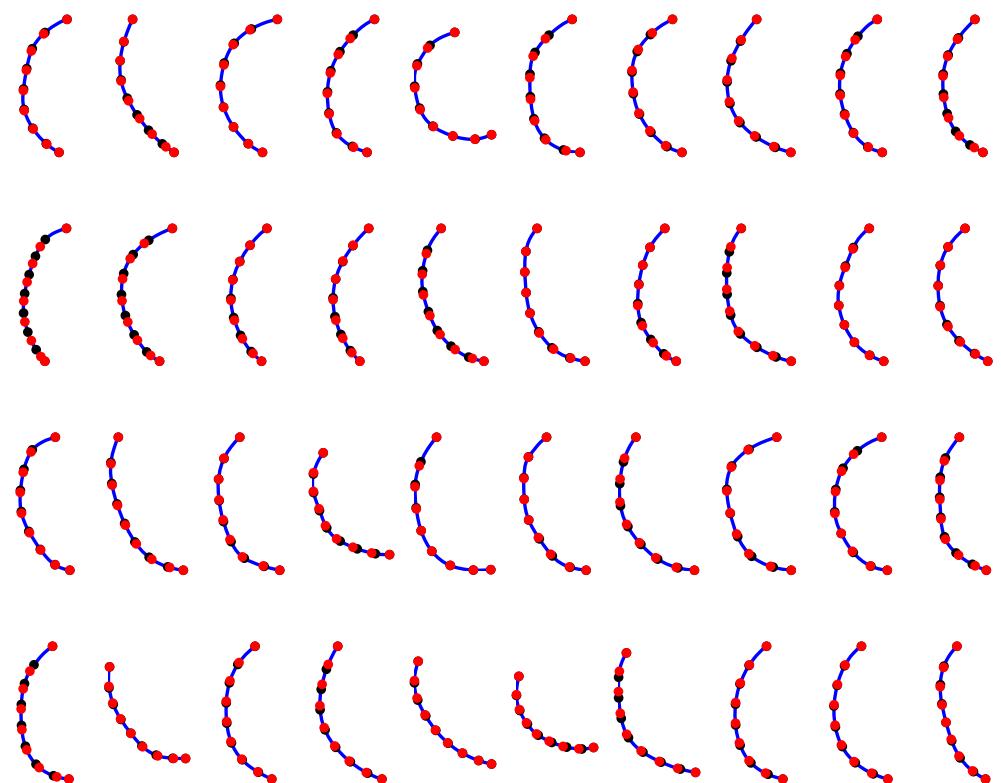
### C.1.2 MDL results for left epicardial contour set



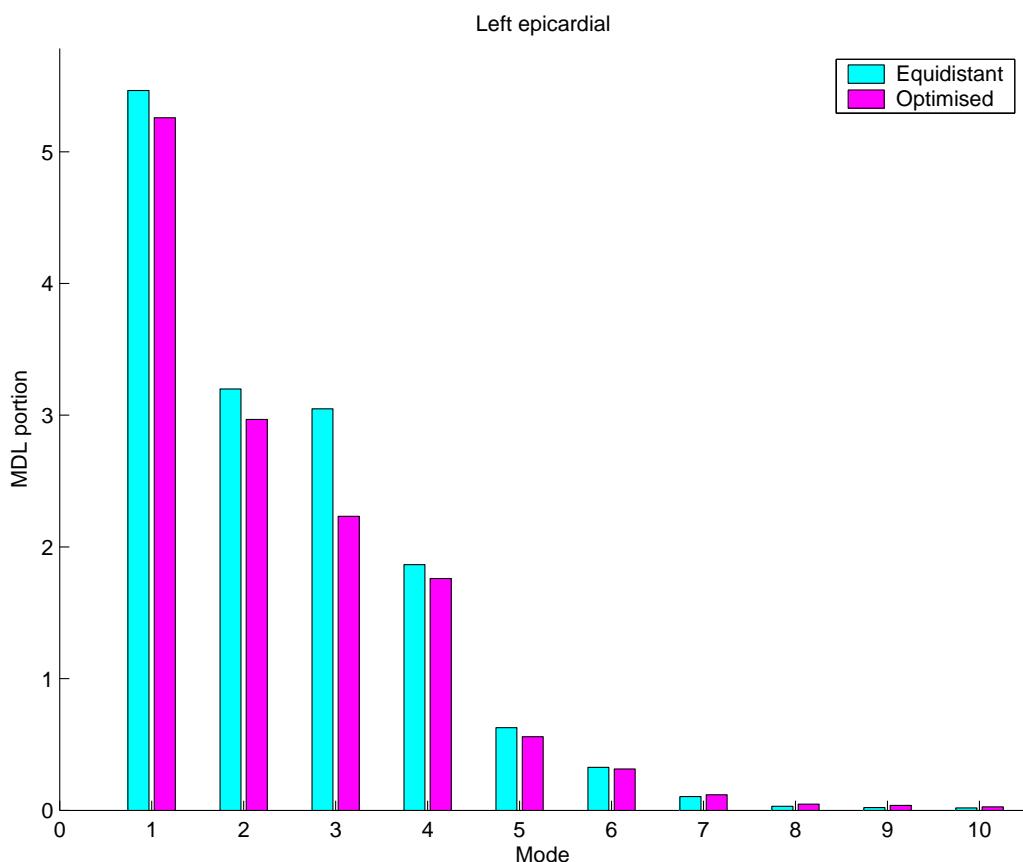
**Figure C.6:** Convergence of MDL optimisation for the left epicardial contour set. Note that the iteration number accounts for  $\#\text{passes} \cdot \#\text{nodes} \cdot \#\text{steps} \cdot \#\text{examples}$ .



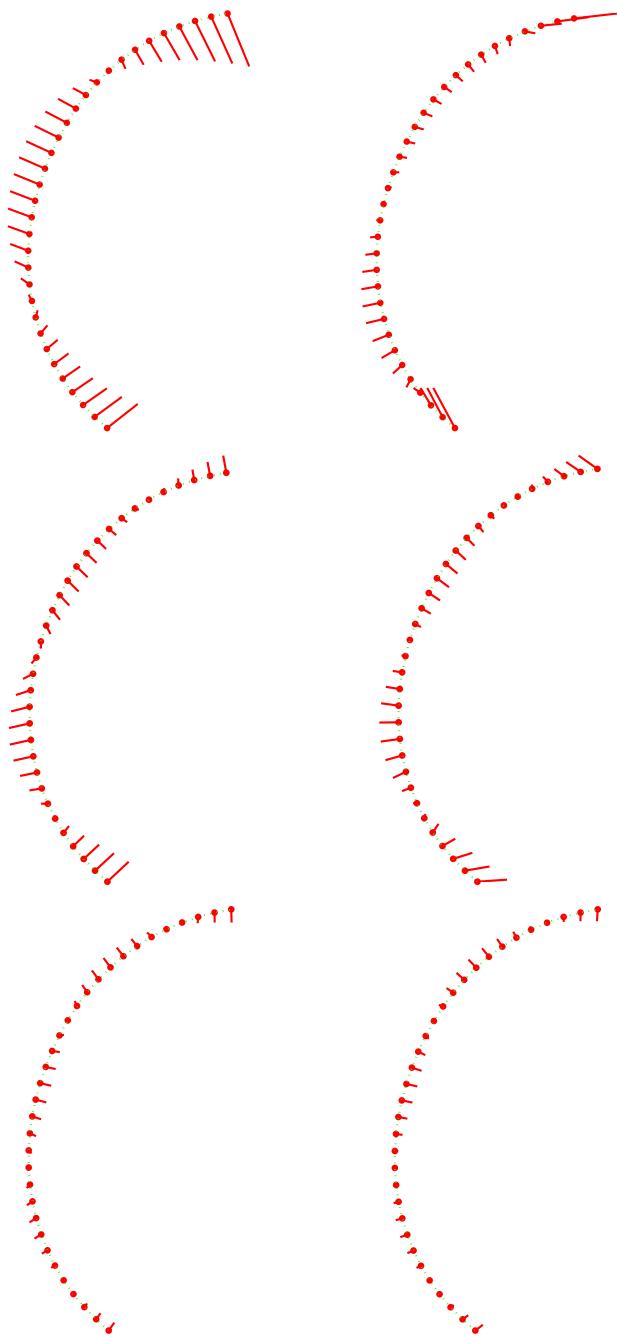
**Figure C.7:** Development of the seven node positions for optimisation of left epicardial contour set. Each node plot shows the value of  $\tau(\text{node})$  plotted versus  $\#\text{passes} \cdot \#\text{steps}$  for all shapes in the set.



**Figure C.8:** Fixed points and node positions for the left epicardial contour set. Equidistant (black) and MDL optimised (red) solution. Each column holds slice 1–4 for each patient.

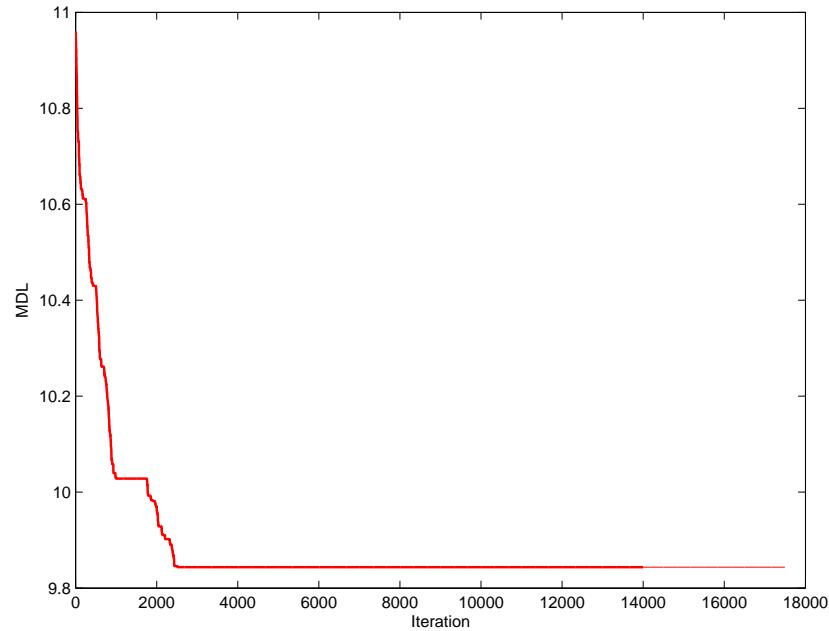


**Figure C.9:** Contribution of the first 10 eigenvalues to the objective function. Comparison of equidistant landmarks and MDL optimised landmarks for the left epicardial set.

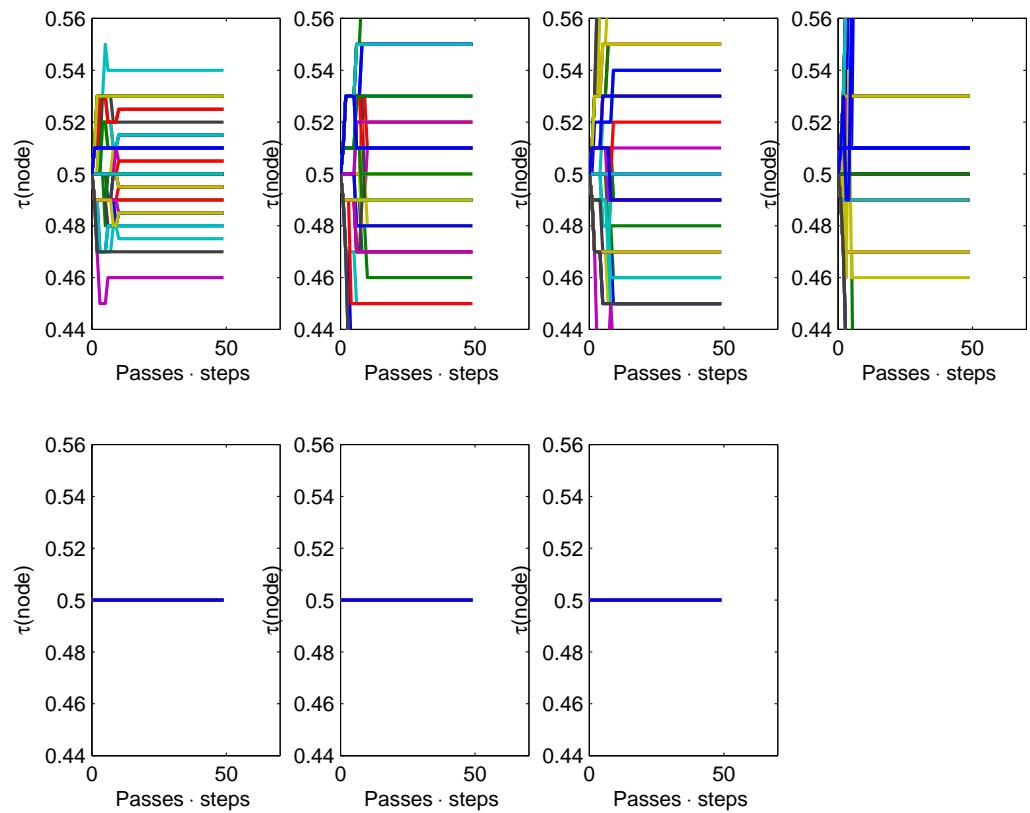


**Figure C.10:** Comparison of principal mode 3,4 and 5 (from top to bottom) before and after optimisation of the left epicardial contour set. Left: Equidistant landmarks, right: MDL optimised landmarks. The whiskers denote 5 standard deviations from mean.

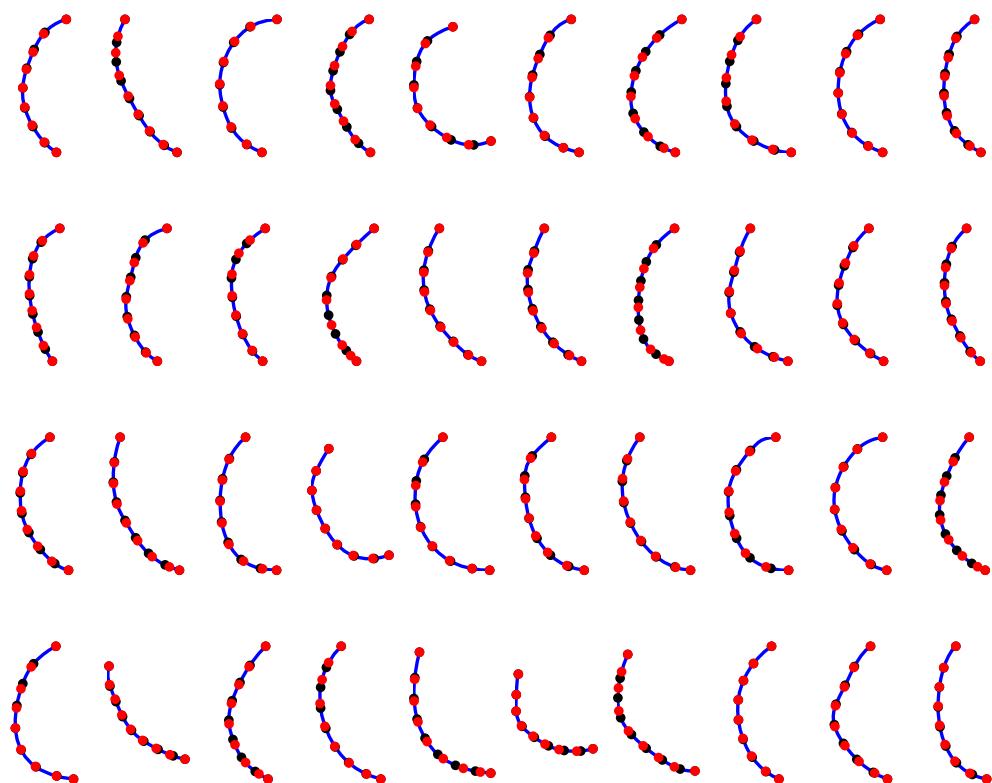
### C.1.3 MDL results for left endocardial contour set



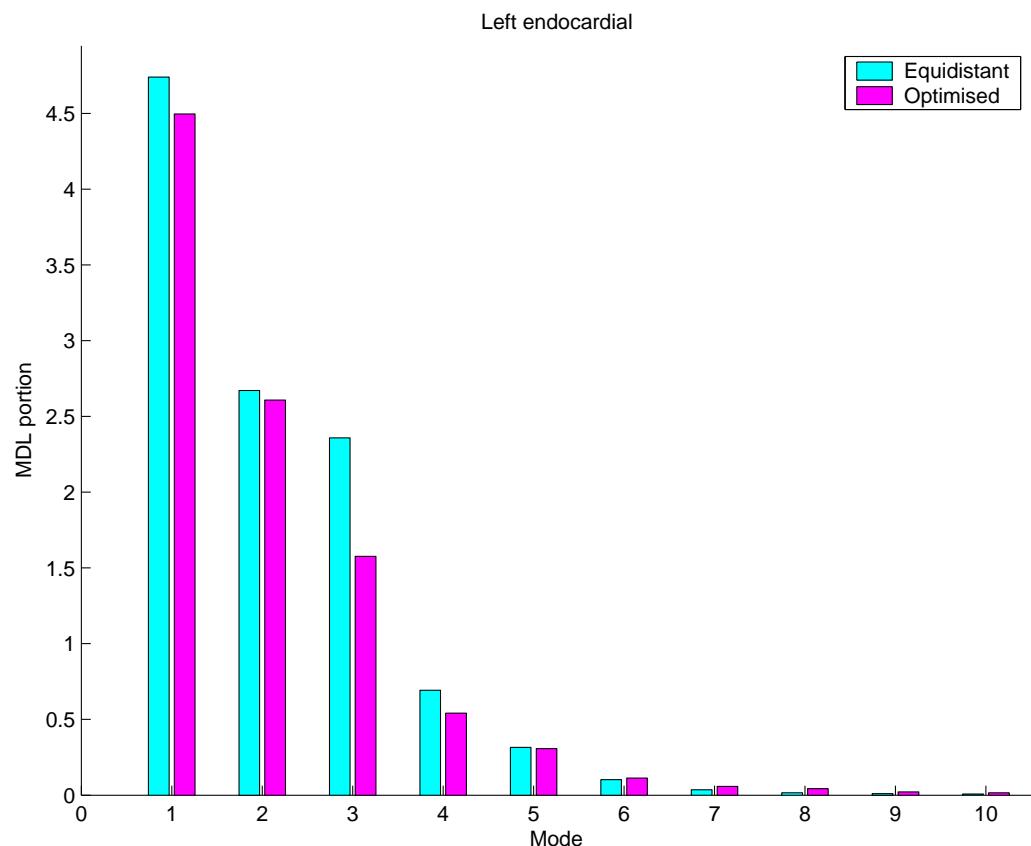
**Figure C.11:** Convergence of MDL optimisation for the left endocardial contour set. Note that the iteration number accounts for #passes · #nodes · #steps · #examples.



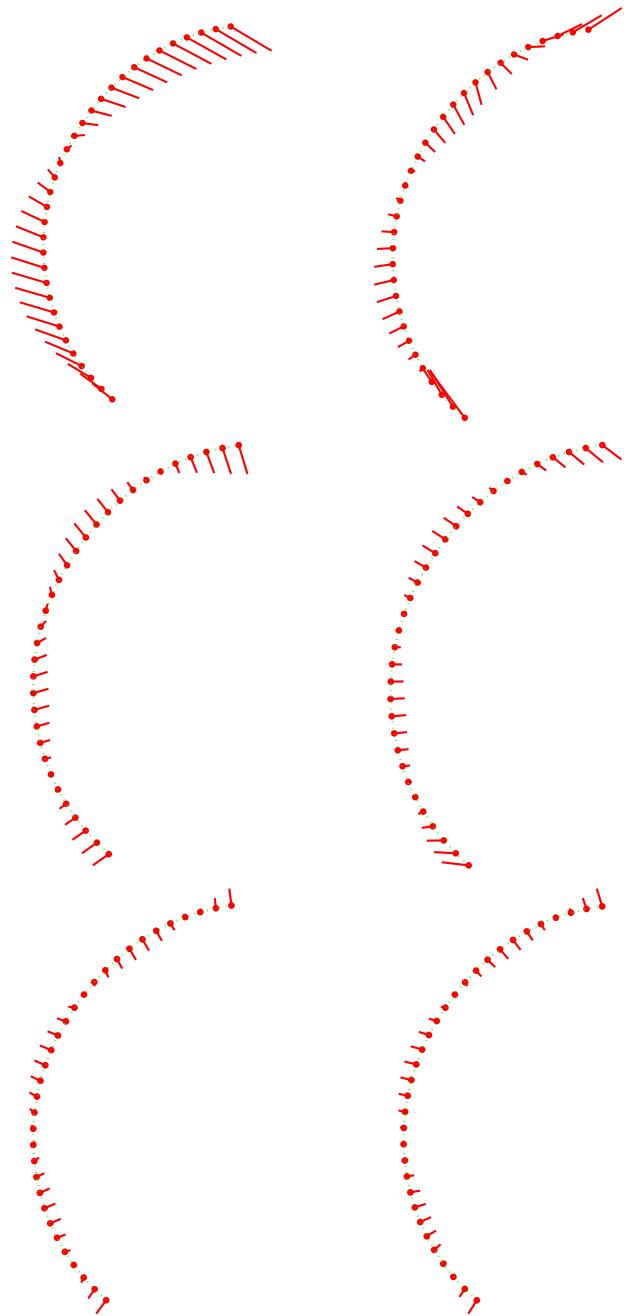
**Figure C.12:** Development of the seven node positions for optimisation of left endocardial contour set. Each node plot shows the value of  $\tau(\text{node})$  plotted versus  $\# \text{passes} \cdot \# \text{steps}$  for all shapes in the set.



**Figure C.13:** Fixed points and node positions for the left endocardial contour set. Equidistant (black) and MDL optimised (red) solution. Each column holds slice 1-4 for each patient.

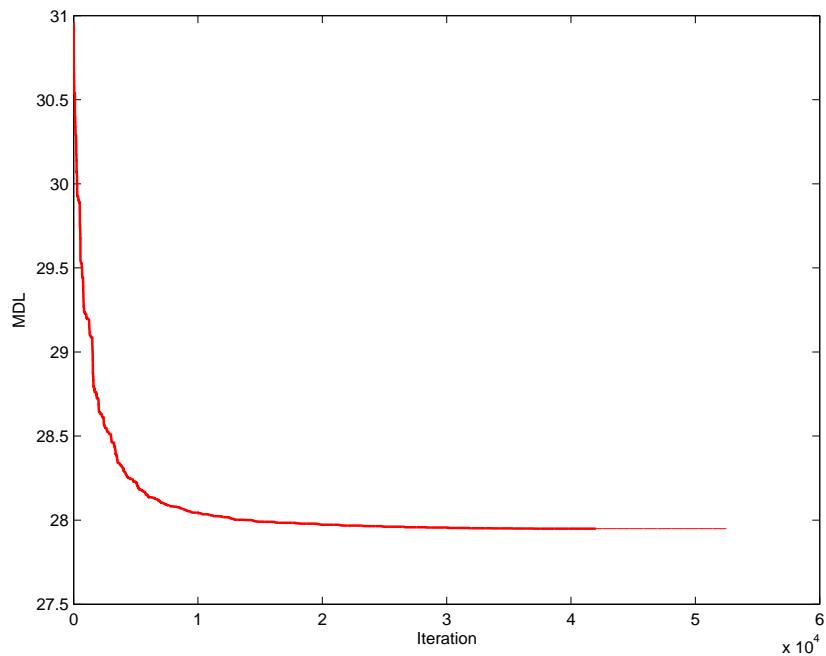


**Figure C.14:** Contribution of the first 10 eigenvalues to the objective function. Comparison of equidistant landmarks and MDL optimised landmarks for the left endocardial set.

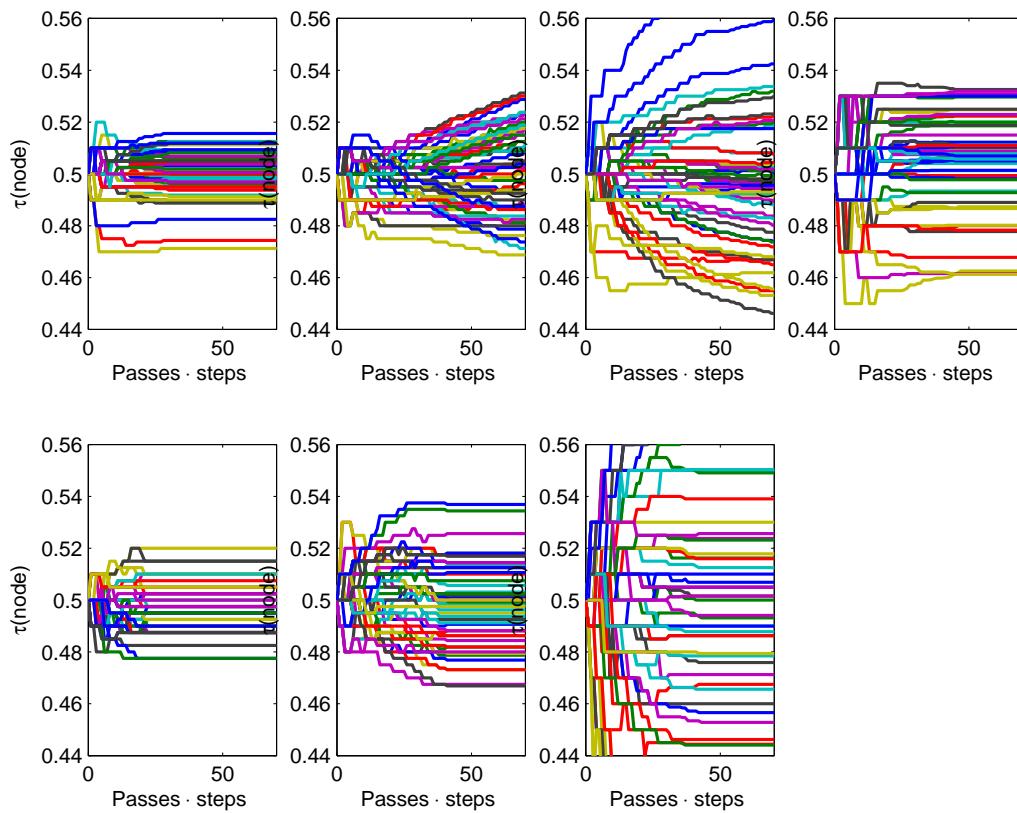


**Figure C.15:** Comparison of principal mode 3,4 and 5 (from top to bottom) before and after optimisation of the left endocardial contour set. Left: Equidistant landmarks, right: MDL optimised landmarks. The whiskers denote 5 standard deviations from mean.

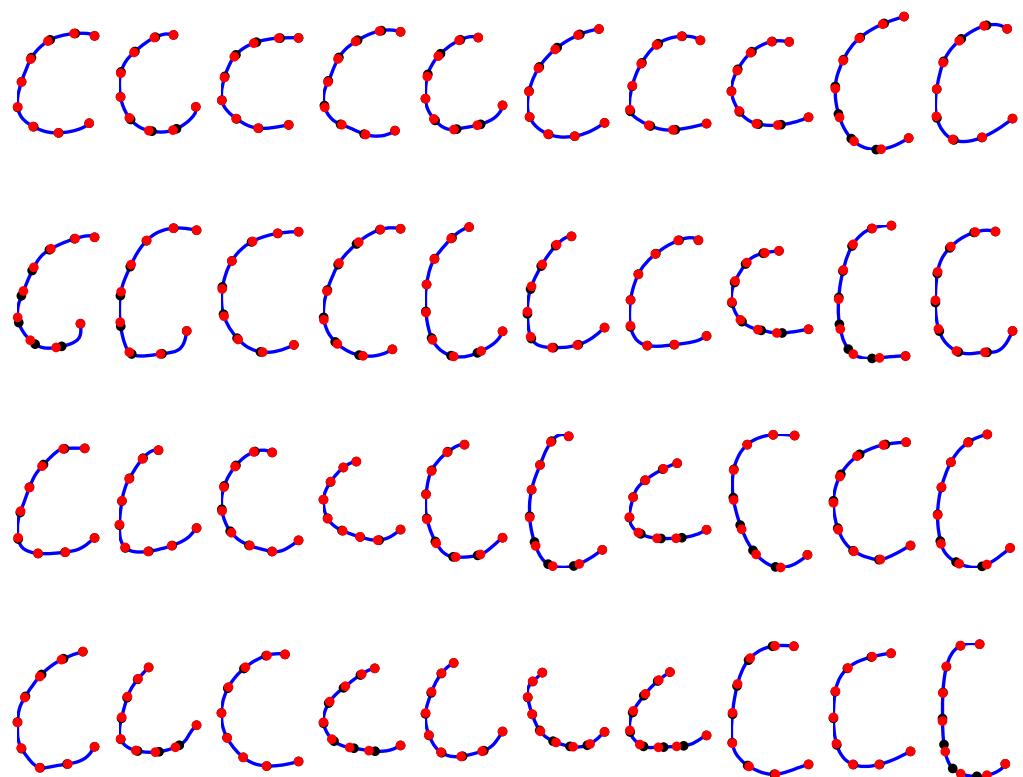
#### C.1.4 MDL results for right ventricle (RV) contour set



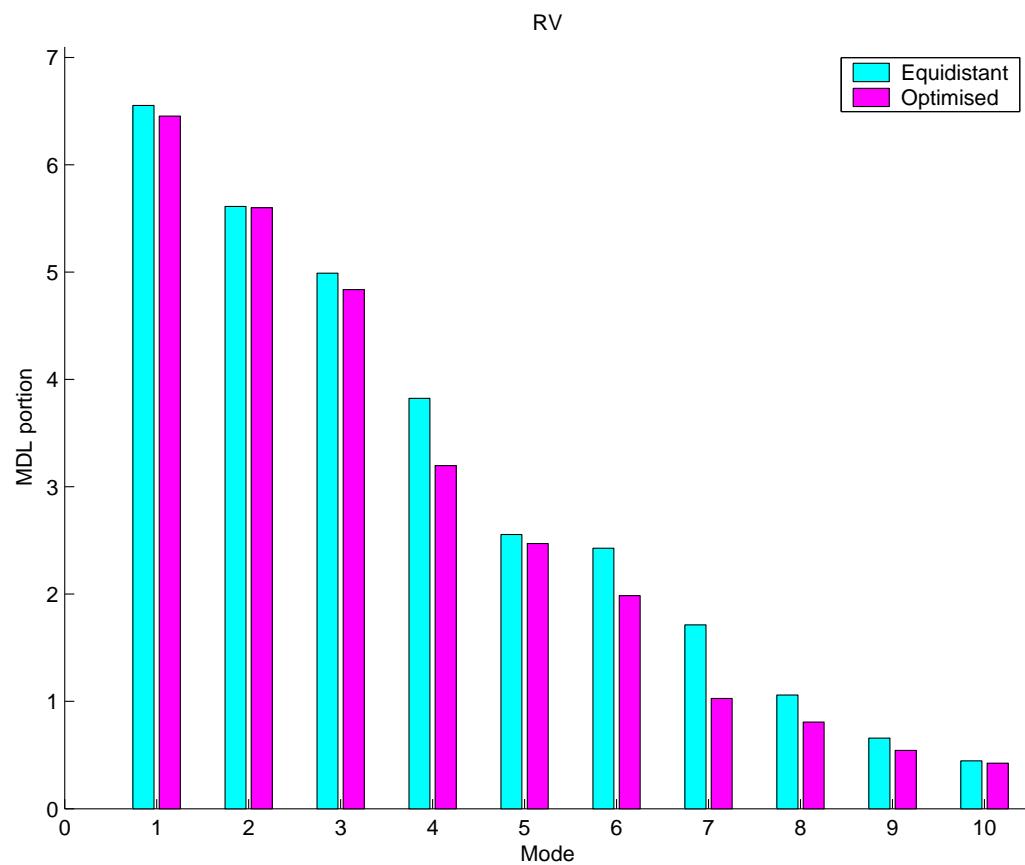
**Figure C.16:** Convergence of MDL optimisation for the RV contour set. Note that the iteration number accounts for  $\# \text{passes} \cdot \# \text{nodes} \cdot \# \text{steps} \cdot \# \text{examples}$ .



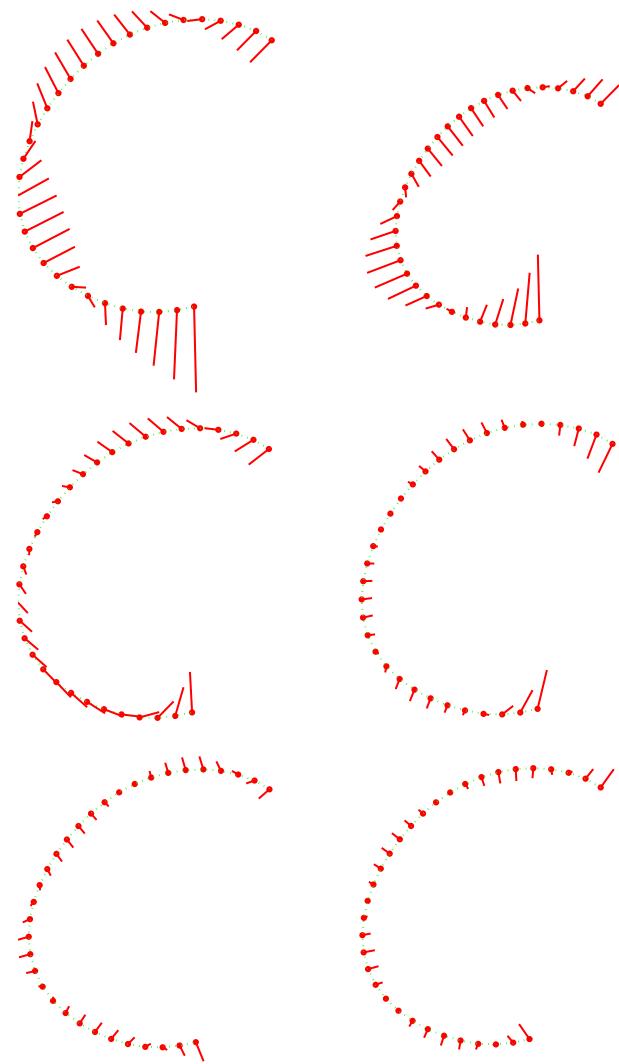
**Figure C.17:** Development of the seven node positions for optimisation of RV contour set. Each node plot shows the value of  $\tau(\text{node})$  plotted versus  $\# \text{passes} \cdot \# \text{steps}$  for all shapes in the set.



**Figure C.18:** Fixed points and node positions for the RV contour set. Equidistant (black) and MDL optimised (red) solution. Each column holds slice 1–4 for each patient.



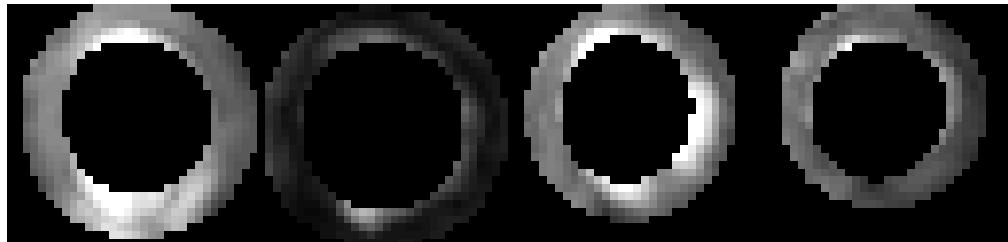
**Figure C.19:** Contribution of the first 10 eigenvalues to the objective function. Comparison of equidistant landmarks and MDL optimised landmarks for the RV set.



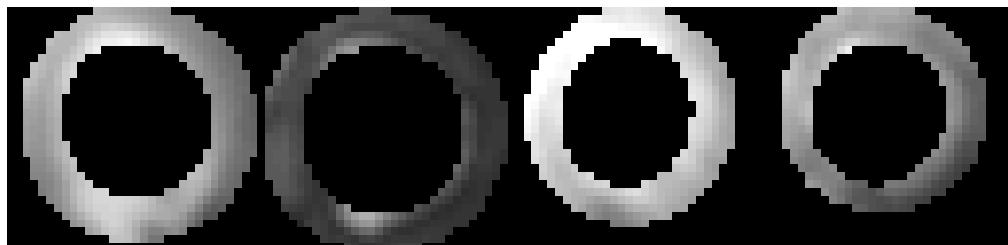
**Figure C.20:** Comparison of principal mode 3,4 and 5 (from top to bottom) before and after optimisation of the RV contour set. Left: Equidistant landmarks, right: MDL optimised landmarks. The whiskers denote 5 standard deviations from mean.

## C.2 Perfusion maps – AAM automatic registration

### C.2.1 Perfusion maps for patient 1



**Figure C.21:** Maximum upslope perfusion maps for patient 1, slice 1–4. Obtained from automatically registered data.

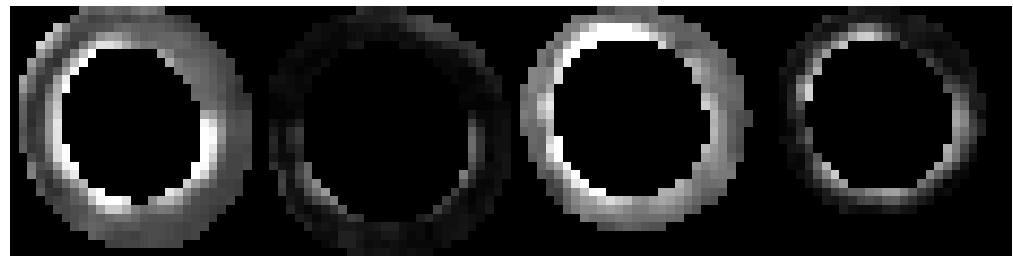


**Figure C.22:** Peak perfusion maps for patient 1, slice 1–4. Obtained from automatically registered data.

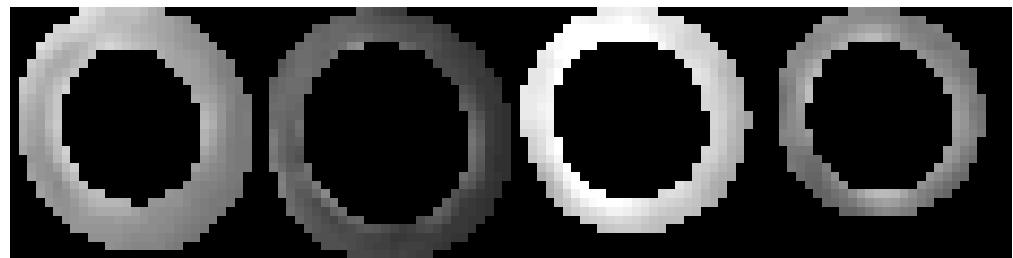


**Figure C.23:** Time-to-peak perfusion maps for patient 1, slice 1–4. Obtained from automatically registered data.

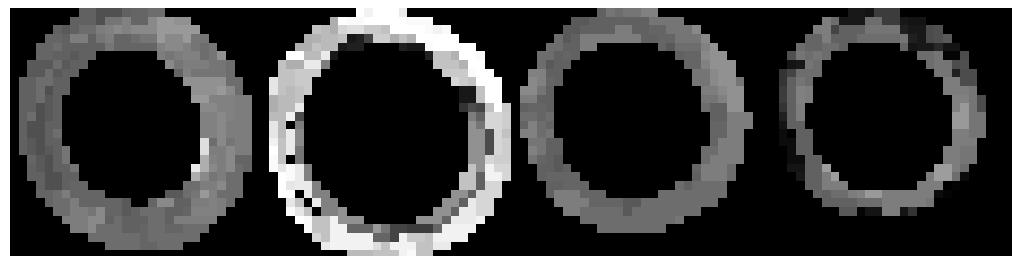
### C.2.2 Perfusion maps for patient 2



**Figure C.24:** Maximum upslope perfusion maps for patient 2, slice 1–4. Obtained from automatically registered data.

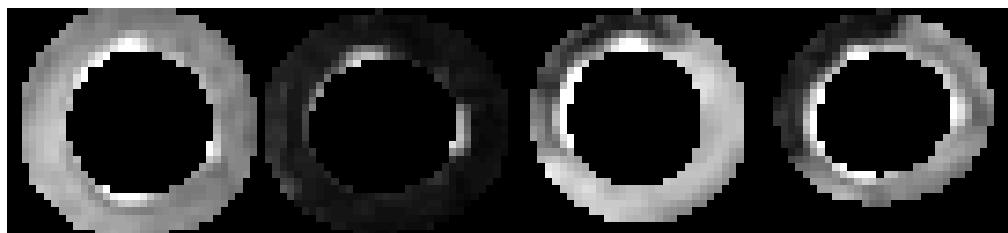


**Figure C.25:** Peak perfusion maps for patient 2, slice 1–4. Obtained from automatically registered data.

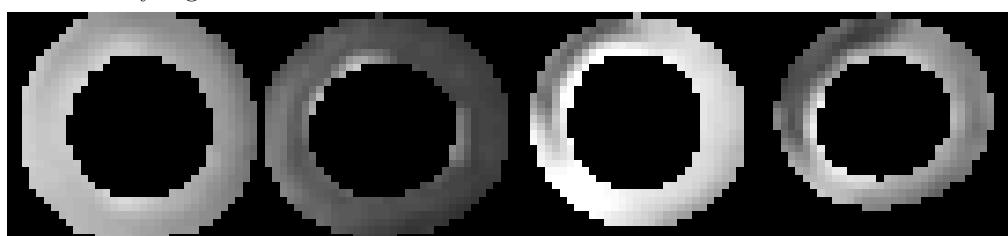


**Figure C.26:** Time-to-peak perfusion maps for patient 2, slice 1–4. Obtained from automatically registered data.

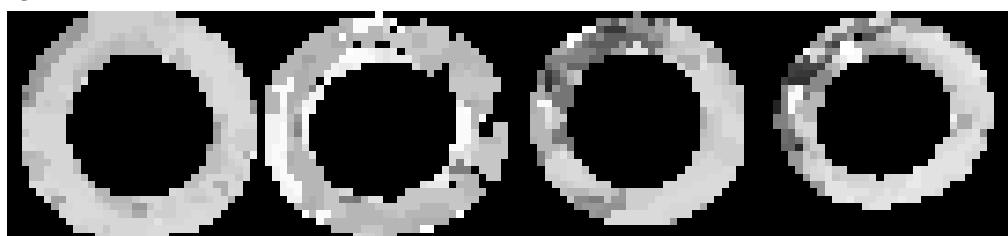
### C.2.3 Perfusion maps for patient 3



**Figure C.27:** Maximum upslope perfusion maps for patient 3, slice 1–4. Obtained from automatically registered data.

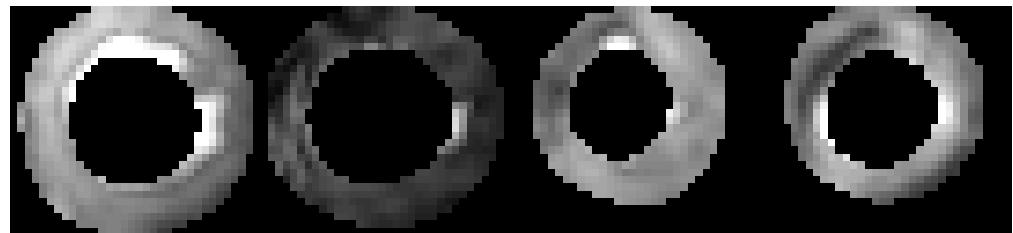


**Figure C.28:** Peak perfusion maps for patient 3, slice 1–4. Obtained from automatically registered data.

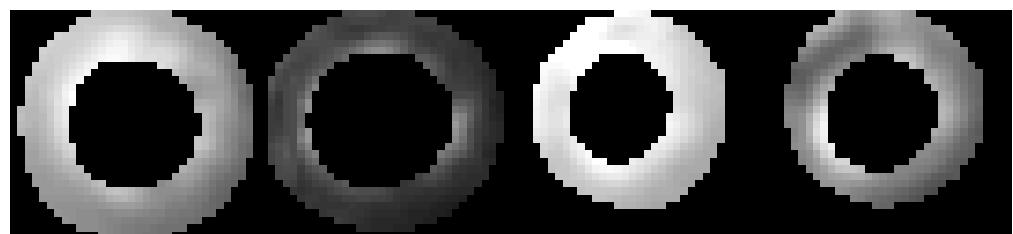


**Figure C.29:** Time-to-peak perfusion maps for patient 3, slice 1–4. Obtained from automatically registered data.

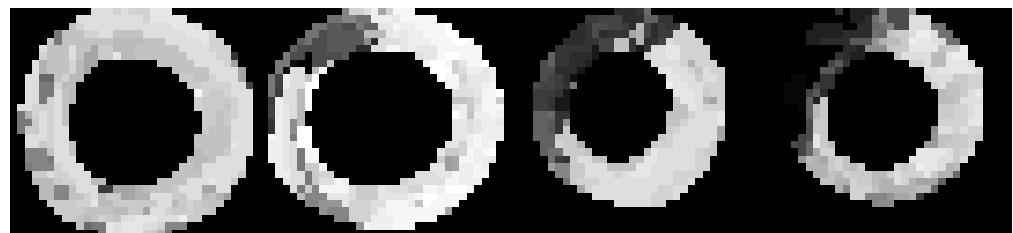
#### C.2.4 Perfusion maps for patient 4



**Figure C.30:** Maximum upslope perfusion maps for patient 4, slice 1–4. Obtained from automatically registered data.

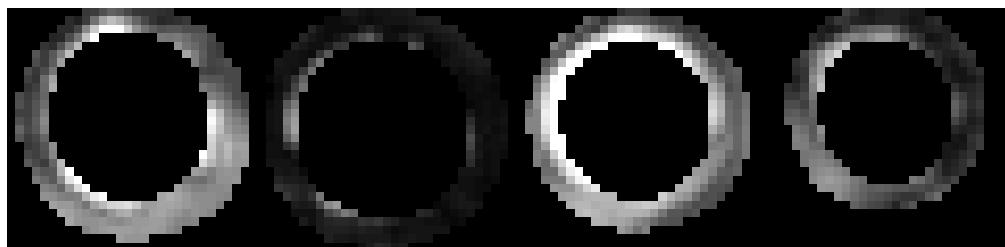


**Figure C.31:** Peak perfusion maps for patient 4, slice 1–4. Obtained from automatically registered data.

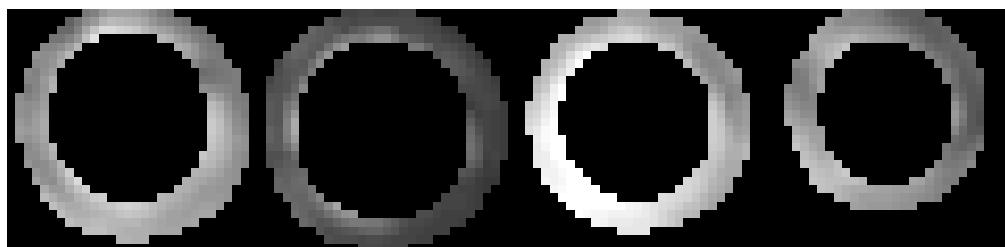


**Figure C.32:** Time-to-peak perfusion maps for patient 4, slice 1–4. Obtained from automatically registered data.

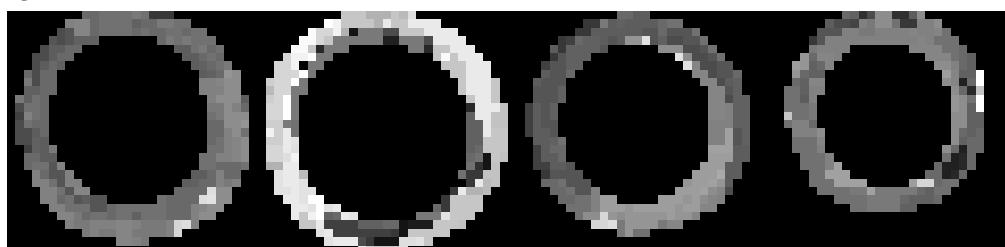
### C.2.5 Perfusion maps for patient 5



**Figure C.33:** Maximum upslope perfusion maps for patient 5, slice 1–4. Obtained from automatically registered data.

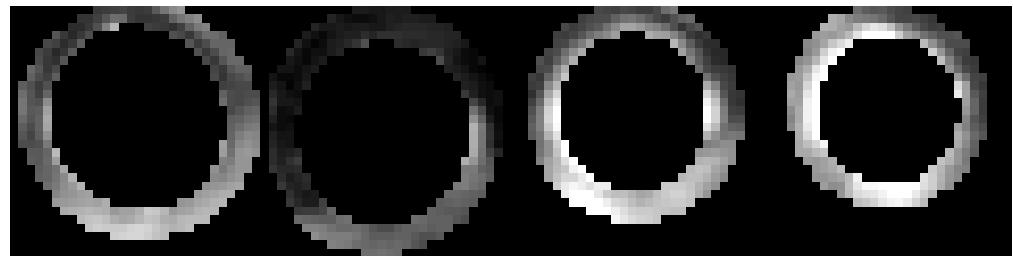


**Figure C.34:** Peak perfusion maps for patient 5, slice 1–4. Obtained from automatically registered data.

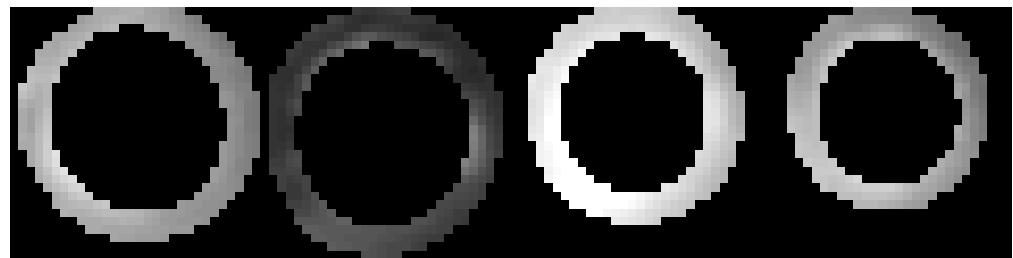


**Figure C.35:** Time-to-peak perfusion maps for patient 5, slice 1–4. Obtained from automatically registered data.

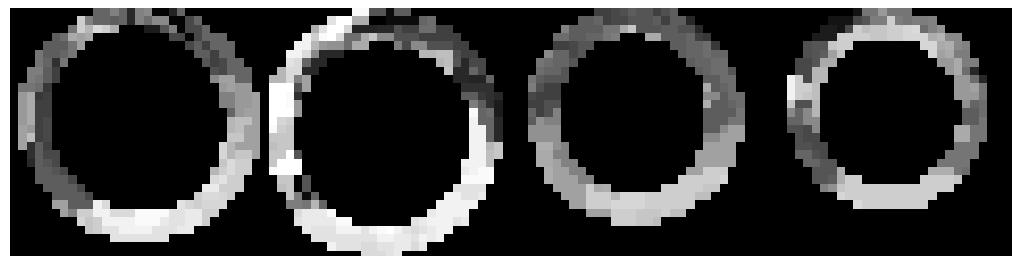
### C.2.6 Perfusion maps for patient 6



**Figure C.36:** Maximum upslope perfusion maps for patient 6, slice 1–4. Obtained from automatically registered data.

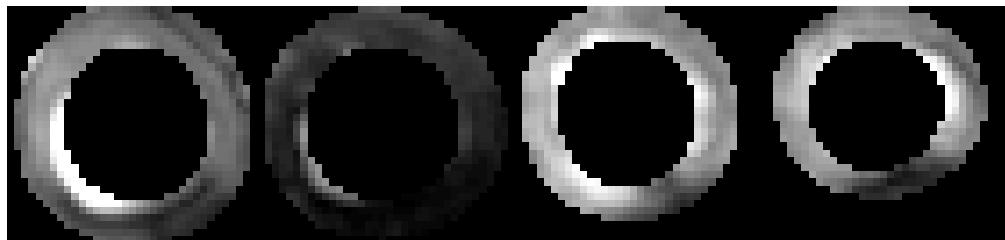


**Figure C.37:** Peak perfusion maps for patient 6, slice 1–4. Obtained from automatically registered data.

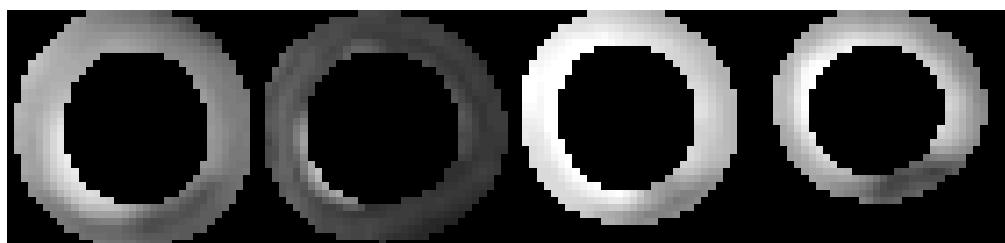


**Figure C.38:** Time-to-peak perfusion maps for patient 6, slice 1–4. Obtained from automatically registered data.

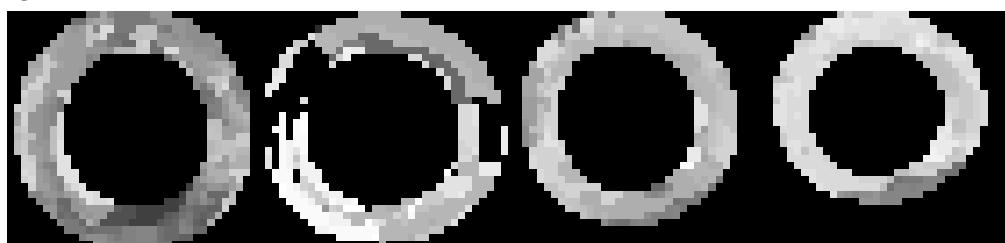
### C.2.7 Perfusion maps for patient 7



**Figure C.39:** Maximum upslope perfusion maps for patient 7, slice 1–4. Obtained from automatically registered data.

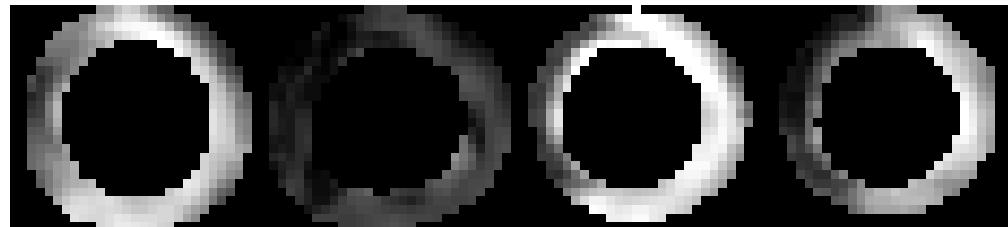


**Figure C.40:** Peak perfusion maps for patient 7, slice 1–4. Obtained from automatically registered data.

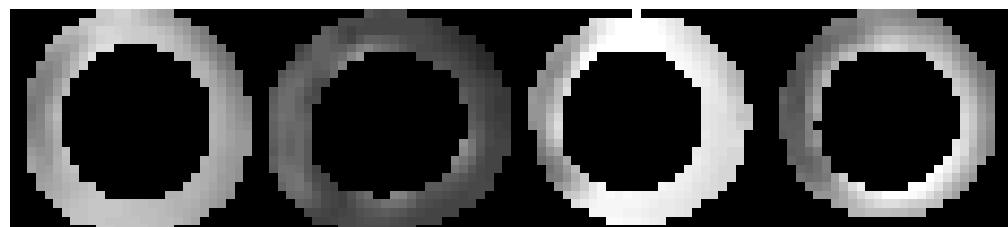


**Figure C.41:** Time-to-peak perfusion maps for patient 7, slice 1–4. Obtained from automatically registered data.

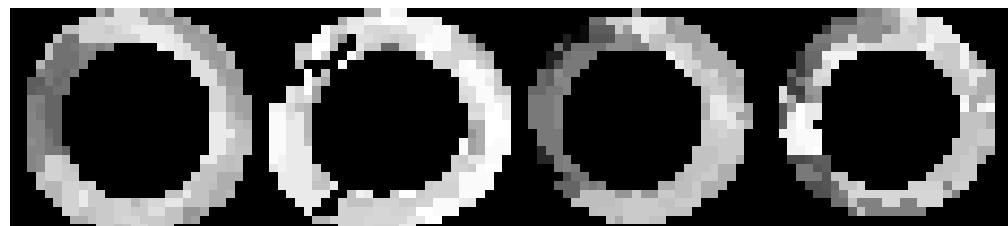
### C.2.8 Perfusion maps for patient 8



**Figure C.42:** Maximum upslope perfusion maps for patient 8, slice 1–4. Obtained from automatically registered data.

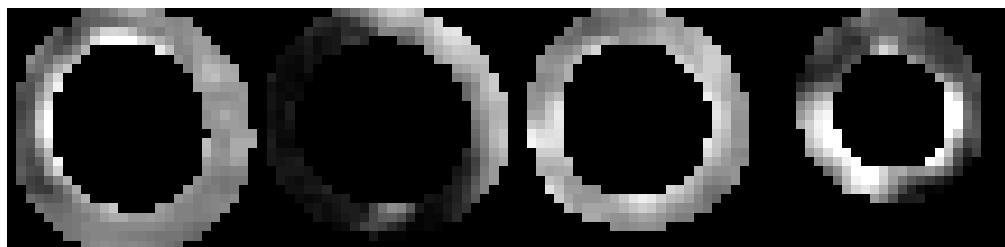


**Figure C.43:** Peak perfusion maps for patient 8, slice 1–4. Obtained from automatically registered data.

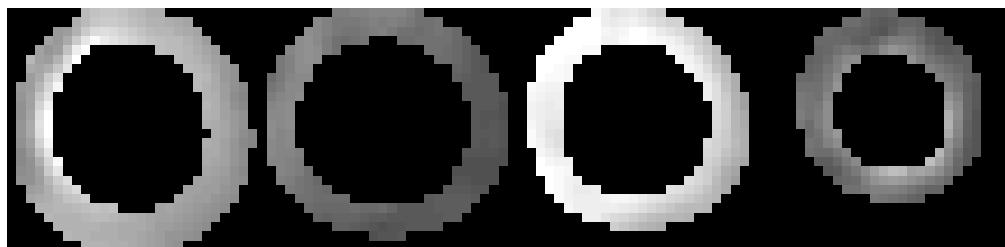


**Figure C.44:** Time-to-peak perfusion maps for patient 8, slice 1–4. Obtained from automatically registered data.

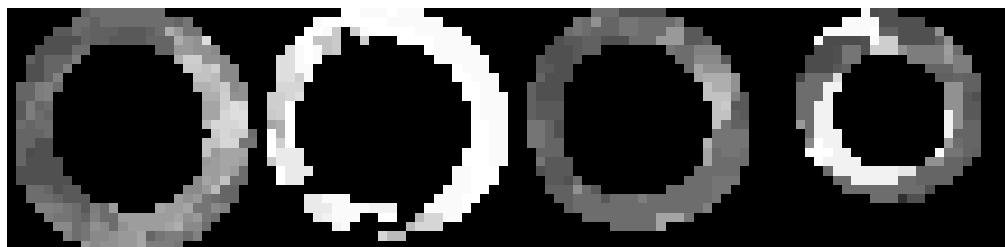
### C.2.9 Perfusion maps for patient 9



**Figure C.45:** Maximum upslope perfusion maps for patient 9, slice 1–4. Obtained from automatically registered data.

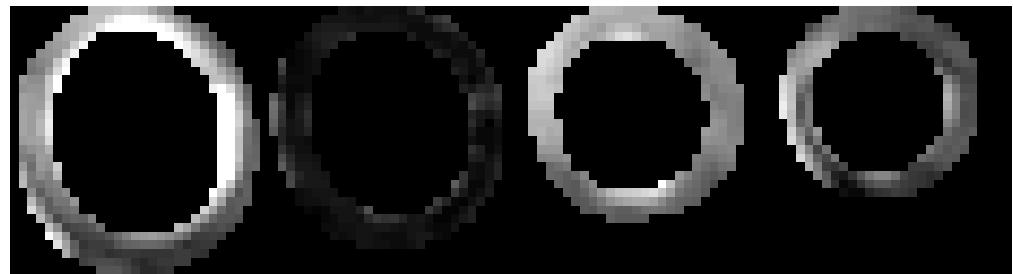


**Figure C.46:** Peak perfusion maps for patient 9, slice 1–4. Obtained from automatically registered data.

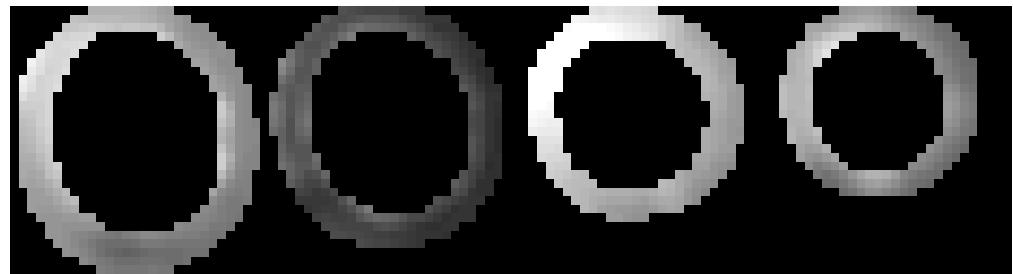


**Figure C.47:** Time-to-peak perfusion maps for patient 9, slice 1–4. Obtained from automatically registered data.

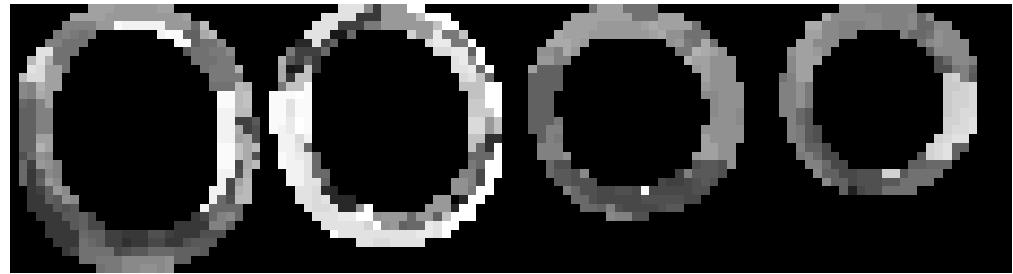
### C.2.10 Perfusion maps for patient 10



**Figure C.48:** Maximum upslope perfusion maps for patient 10, slice 1–4. Obtained from automatically registered data.



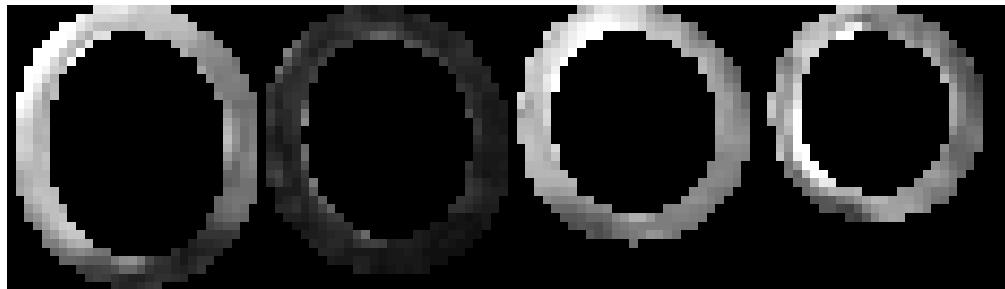
**Figure C.49:** Peak perfusion maps for patient 10, slice 1–4. Obtained from automatically registered data.



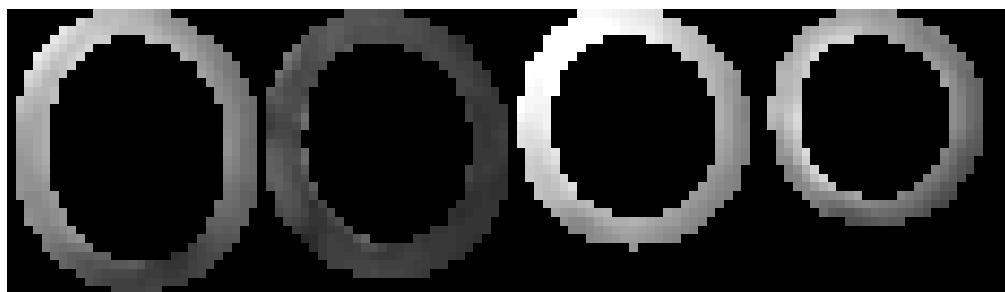
**Figure C.50:** Time-to-peak perfusion maps for patient 10, slice 1–4. Obtained from automatically registered data.

### C.3 Perfusion maps – Ground truth

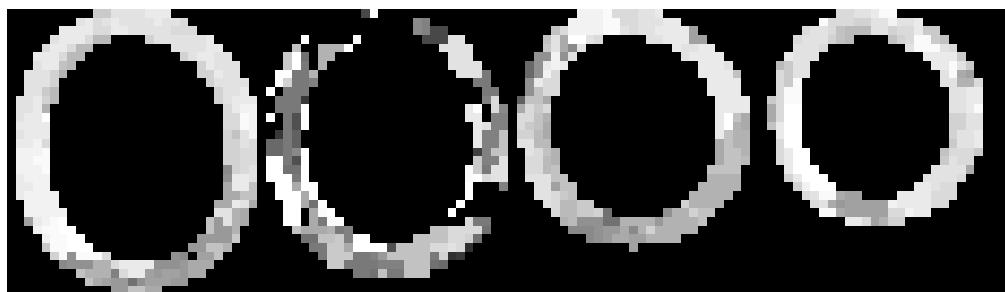
#### C.3.1 Perfusion maps for patient 1



**Figure C.51:** Maximum upslope perfusion maps for patient 1, slice 1–4. Obtained from ground truth data.

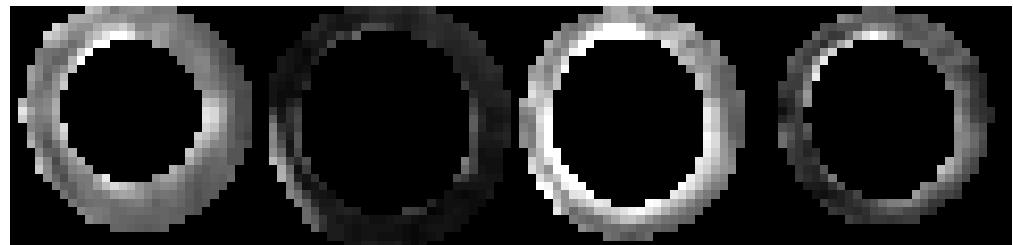


**Figure C.52:** Peak perfusion maps for patient 1, slice 1–4. Obtained from ground truth data.

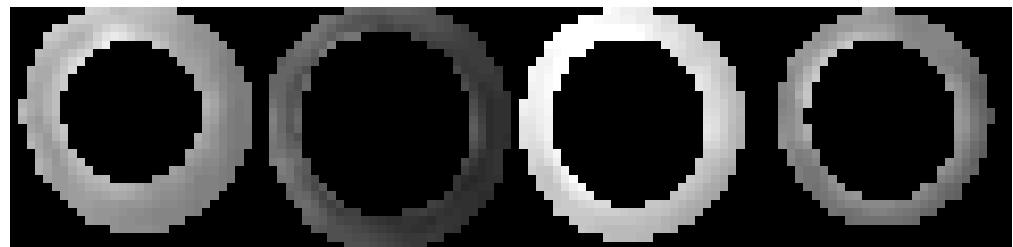


**Figure C.53:** Time-to-peak perfusion maps for patient 1, slice 1–4. Obtained from ground truth data.

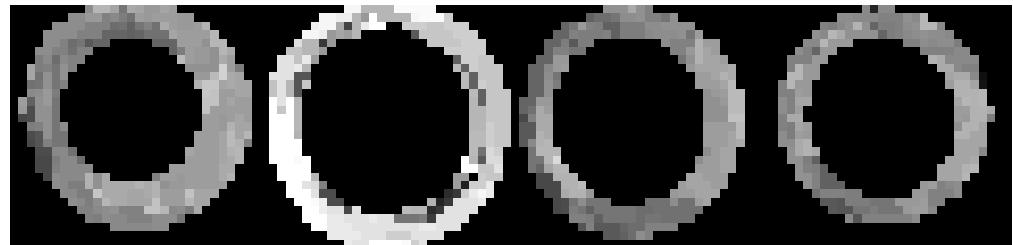
### C.3.2 Perfusion maps for patient 2



**Figure C.54:** Maximum upslope perfusion maps for patient 2, slice 1–4. Obtained from ground truth data.

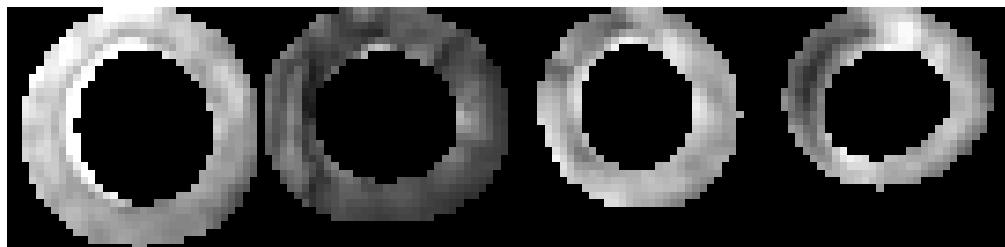


**Figure C.55:** Peak perfusion maps for patient 2, slice 1–4. Obtained from ground truth data.

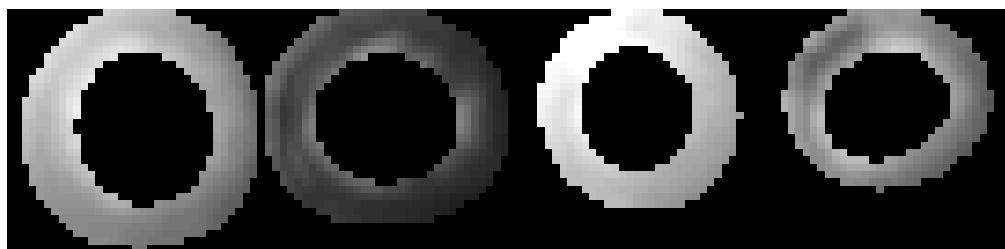


**Figure C.56:** Time-to-peak perfusion maps for patient 2, slice 1–4. Obtained from ground truth data.

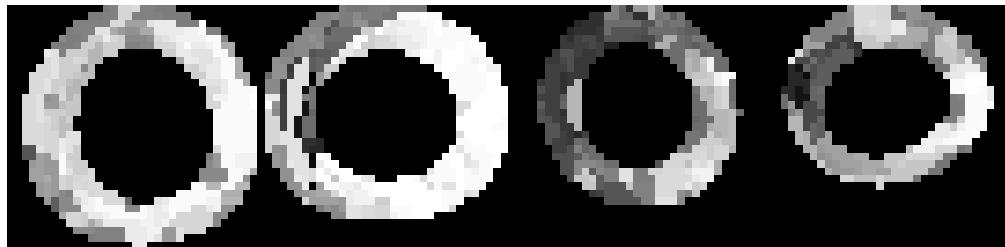
### C.3.3 Perfusion maps for patient 3



**Figure C.57:** Maximum upslope perfusion maps for patient 3, slice 1–4. Obtained from ground truth data.

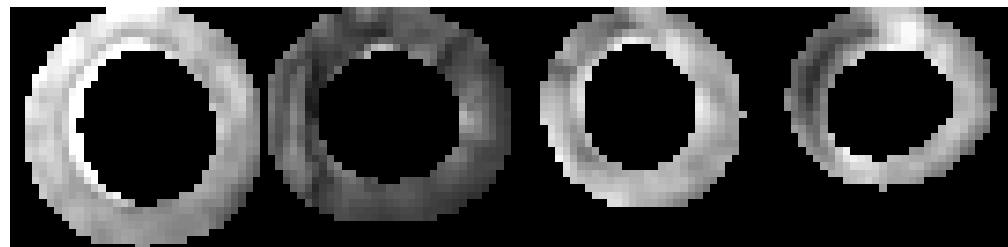


**Figure C.58:** Peak perfusion maps for patient 3, slice 1–4. Obtained from ground truth data.

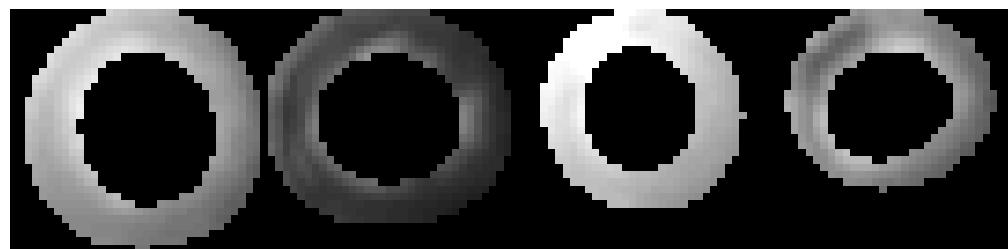


**Figure C.59:** Time-to-peak perfusion maps for patient 3, slice 1–4. Obtained from ground truth data.

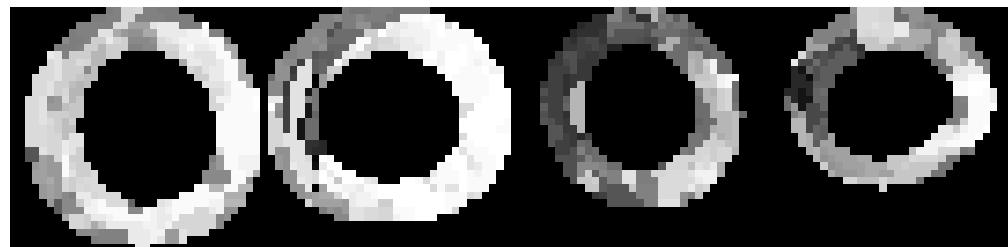
### C.3.4 Perfusion maps for patient 4



**Figure C.60:** Maximum upslope perfusion maps for patient 4, slice 1–4. Obtained from ground truth data.

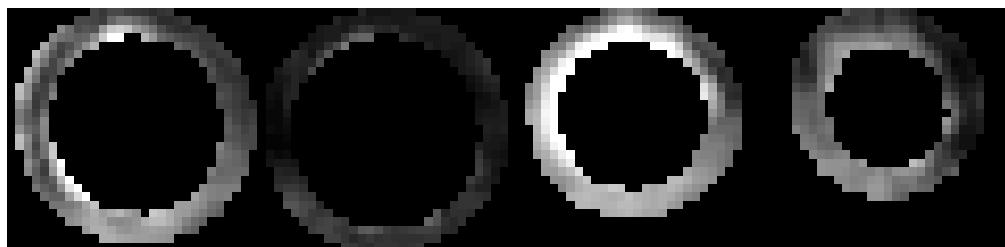


**Figure C.61:** Peak perfusion maps for patient 4, slice 1–4. Obtained from ground truth data.

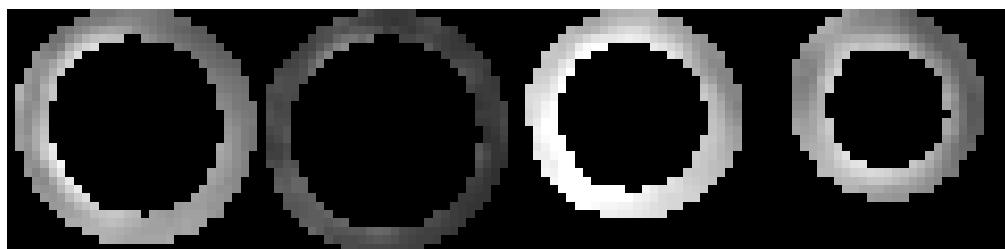


**Figure C.62:** Time-to-peak perfusion maps for patient 4, slice 1–4. Obtained from ground truth data.

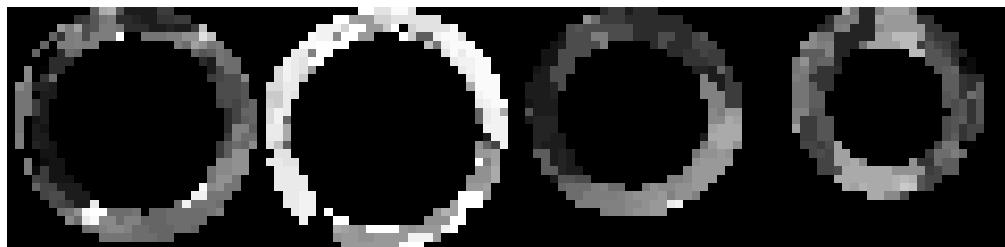
### C.3.5 Perfusion maps for patient 5



**Figure C.63:** Maximum upslope perfusion maps for patient 5, slice 1–4. Obtained from ground truth data.

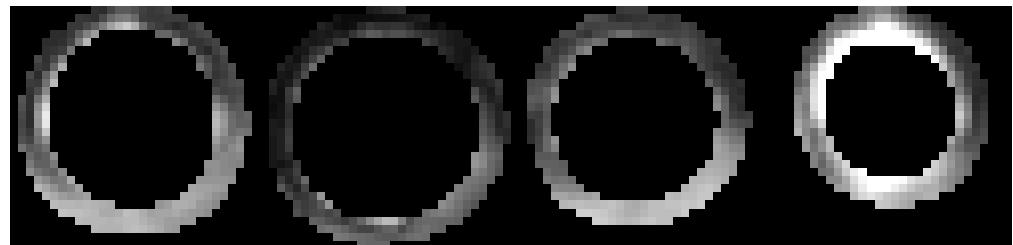


**Figure C.64:** Peak perfusion maps for patient 5, slice 1–4. Obtained from ground truth data.

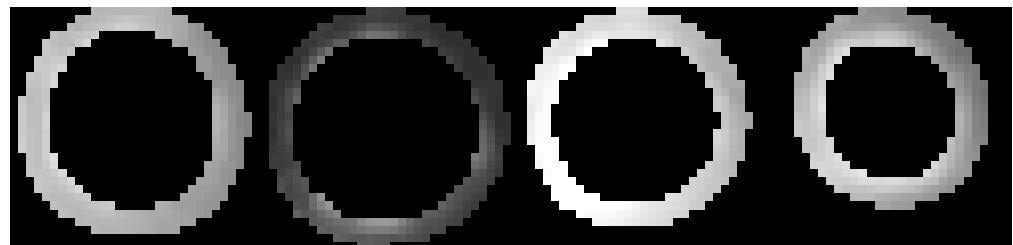


**Figure C.65:** Time-to-peak perfusion maps for patient 5, slice 1–4. Obtained from ground truth data.

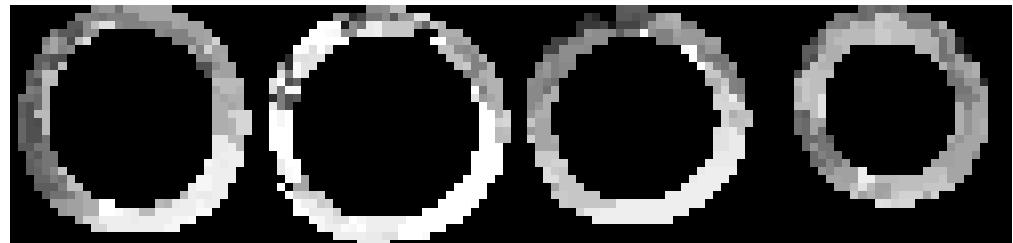
### C.3.6 Perfusion maps for patient 6



**Figure C.66:** Maximum upslope perfusion maps for patient 6, slice 1–4. Obtained from ground truth data.

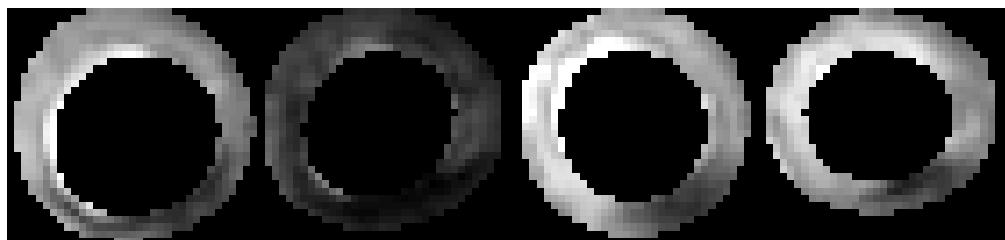


**Figure C.67:** Peak perfusion maps for patient 6, slice 1–4. Obtained from ground truth data.

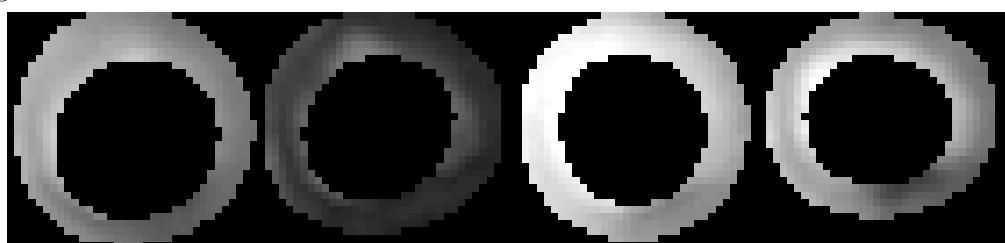


**Figure C.68:** Time-to-peak perfusion maps for patient 6, slice 1–4. Obtained from ground truth data.

### C.3.7 Perfusion maps for patient 7



**Figure C.69:** Maximum upslope perfusion maps for patient 7, slice 1–4. Obtained from ground truth data.

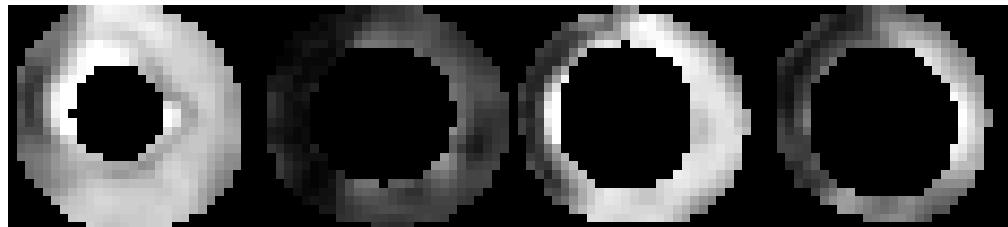


**Figure C.70:** Peak perfusion maps for patient 7, slice 1–4. Obtained from ground truth data.

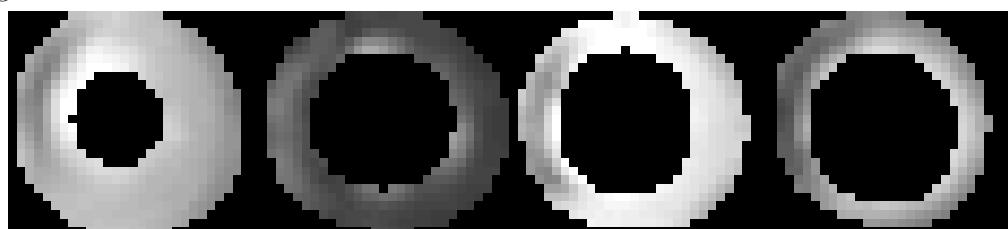


**Figure C.71:** Time-to-peak perfusion maps for patient 7, slice 1–4. Obtained from ground truth data.

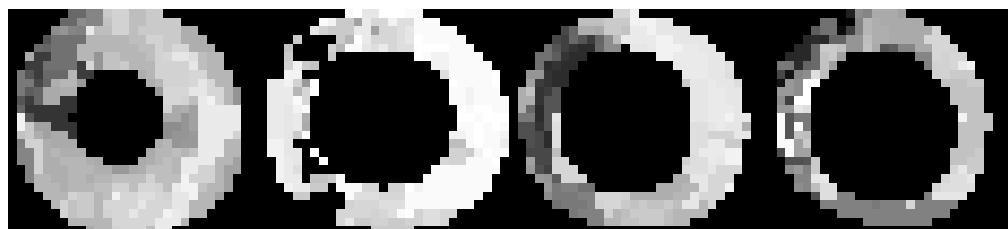
### C.3.8 Perfusion maps for patient 8



**Figure C.72:** Maximum upslope perfusion maps for patient 8, slice 1–4. Obtained from ground truth data.

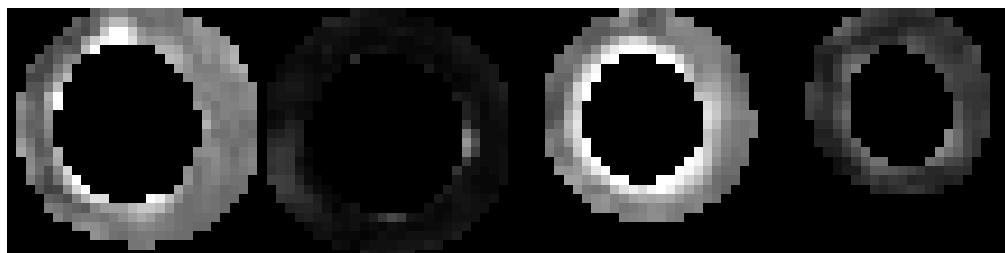


**Figure C.73:** Peak perfusion maps for patient 8, slice 1–4. Obtained from ground truth data.

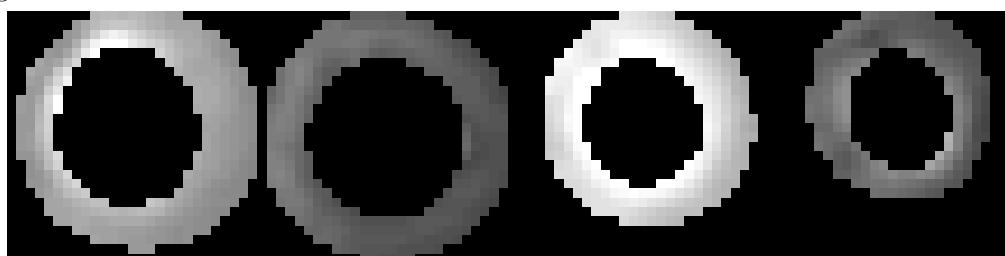


**Figure C.74:** Time-to-peak perfusion maps for patient 8, slice 1–4. Obtained from ground truth data.

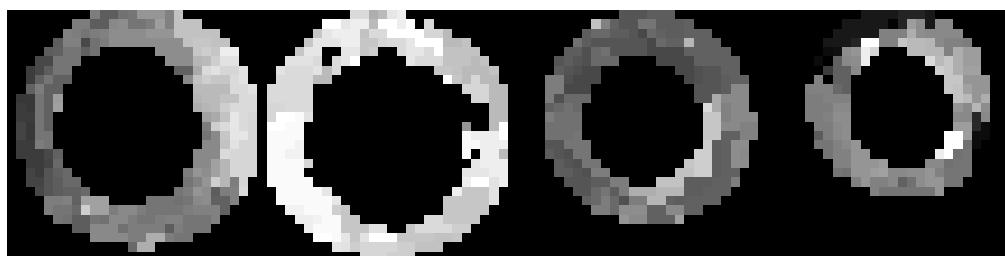
### C.3.9 Perfusion maps for patient 9



**Figure C.75:** Maximum upslope perfusion maps for patient 9, slice 1–4. Obtained from ground truth data.

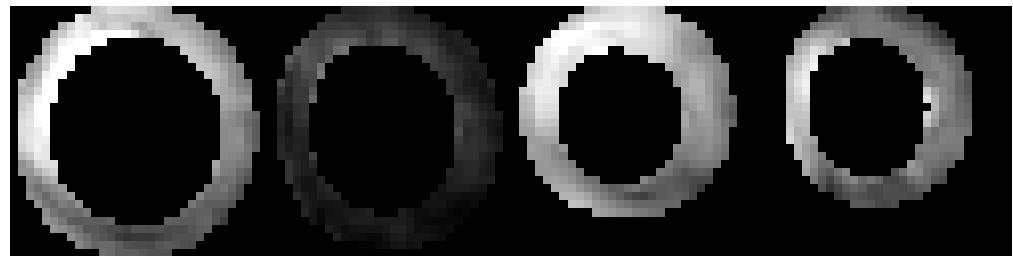


**Figure C.76:** Peak perfusion maps for patient 9, slice 1–4. Obtained from ground truth data.

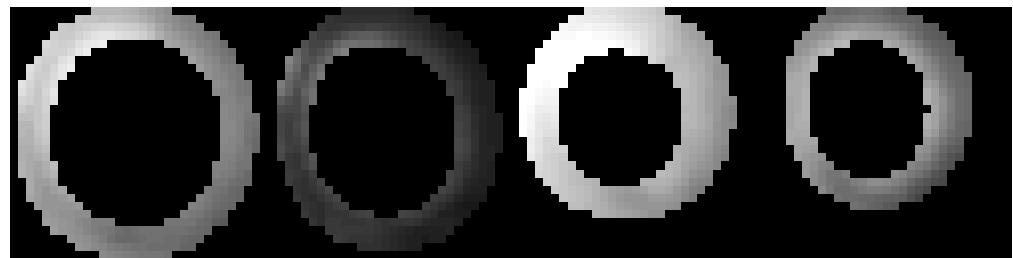


**Figure C.77:** Time-to-peak perfusion maps for patient 9, slice 1–4. Obtained from ground truth data.

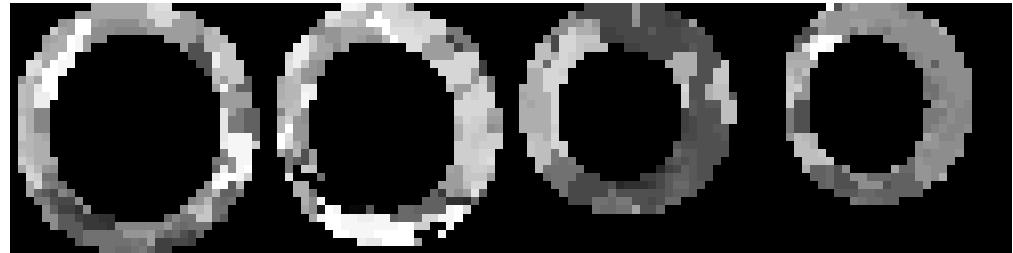
### C.3.10 Perfusion maps for patient 10



**Figure C.78:** Maximum upslope perfusion maps for patient 10, slice 1–4. Obtained from ground truth data.



**Figure C.79:** Peak perfusion maps for patient 10, slice 1–4. Obtained from ground truth data.



**Figure C.80:** Time-to-peak perfusion maps for patient 10, slice 1–4. Obtained from ground truth data.