CHAPTER 7

Time continuous tracking and segmentation of cardiovascular magnetic resonance images using multi-dimensional dynamic programming

This chapter was adapted from:

Time continuous tracking and segmentation of cardiovascular magnetic resonance images using multi-dimensional dynamic programming
Mehmet Üzümcü, Rob J. van der Geest, Cory Swingen, Johan H.C. Reiber, Boudewijn P.F. Lelieveldt
ABSTRACT

In this article, we propose a semiautomatic method for time-continuous contour detection in all phases of the cardiac cycle in magnetic resonance sequences. The method is based on multidimensional dynamic programming. After shape parameterization, cost hypercubes are filled with image-feature derived cost function values. Using multidimensional dynamic programming, an optimal path is sought through the sequence of hypercubes. Constraints can be imposed by setting limits to the parameter changes between subsequent hypercubes. Quantitative evaluation was performed on 20 subjects. Average border positioning error over all slices, all phases and all studies, was 1.77 ± 0.57 mm for epicardial and 1.86 ± 0.59 mm for endocardial contours. The average error in end-diastolic and end-systolic volumes over all studies was small: 4.24 ± 4.62 ml and -4.36 ± 4.26 ml, respectively. The average error in ejection fraction was 4.82 ± 3.01%. The reported results compare favorable to the best reported results in recent literature, underlining the potential of this method for application in daily clinical practice.
7.1 Introduction

For quantitative analysis of cardiac function, clinical parameters such as end-diastolic (ED) and end-systolic (ES) volume, ejection fraction (EF), and ventricular wall motion are relevant. These parameters are usually estimated from short axis acquisitions consisting of 200 to 300 images. To quantitatively analyze global and regional cardiac function, endo- and epicardial contours of the cardiac left ventricle are required. Recently, several (semi-)automatic methods were developed for detecting the contours of the myocardium. Van Assen et al.\textsuperscript{1} and Lötjönen et al.\textsuperscript{2} have proposed 3-dimensional (3D) active shape models for automatic segmentation of the human cardiac left ventricle. Kaus et al.\textsuperscript{3} have developed deformable models constrained by prior knowledge for automatic segmentation of the left ventricle. Sanchez-Ortiz et al.\textsuperscript{4} propose a multi-scale fuzzy clustering-based segmentation algorithm for contour detection in 3D cardiac echocardiographic images. However, many of the proposed methods are suitable for segmentation in end-diastolic and end-systolic phases only, e.g., Lorenzo-Valdés et al.\textsuperscript{8} have developed a 4D probabilistic atlas for the 4D (3D + time) segmentation of the left ventricle based on an expectation maximization method. This method yields segmentations over the full cardiac cycle. However, many manually segmented images are needed for building the atlas, and the structures are segmented independently in each timeframe. Bosch et al.\textsuperscript{9} and van der Geest et al.\textsuperscript{10} have demonstrated that active appearance motion models are able to yield time-continuous segmentation results for 2D echocardiographic and cardiac magnetic resonance (MR) image sequences, respectively. This method, however, needs to be trained on a training set consisting of several cardiac MR scans with manually annotated contours in all slices and all frames. For a complete overview of recent developments in cardiac segmentation techniques, see the study by Frangi et al.\textsuperscript{11}.

The mentioned methods here either do not yield time continuous segmentation results over the full cardiac cycle or require extensive manually annotated datasets for training. The goal of this work is to develop a method for time-continuous segmentation of the full cardiac cycle that does not require an extensive training set. To accomplish this, we propose an extension of the well-known 2D dynamic programming to higher dimensions. By expressing a shape with a limited number of parameters, cost hypercubes are constructed, in which each axis represents a parameter range. Each node in a hypercube represents a shape instantiation, and a connective path through a sequence of hypercubes represents a changing shape in a time sequence of images. Constraints
such as temporal or spatial continuity are imposed by setting limits to the parameter changes for each axis between subsequent hypercubes. This makes the method robust in the presence of artifacts or missing image information in individual frames. By including information from all frames in the segmentation, we expect the method to yield more robust and consistent contours in ED and ES frames, which are generally used for deriving the parameters that are needed for quantitative analysis of the cardiac left ventricle. Furthermore, other clinically relevant time-dependent parameters can be derived from the segmented full cycle such as regional wall motion, rate of wall thickening, and peak ejection/filling rate. The proposed method does not require training on large datasets, and prior knowledge on global shape or position dynamics is not needed.

7.2 Materials and Methods

7.2.1 Background
Dynamic programming\textsuperscript{13} is a method for solving variational problems by successively selecting the locally optimal solutions. The Dijkstra algorithm\textsuperscript{13} is one of the best known dynamic programming algorithms and is used for solving shortest path problems, i.e., finding the optimal path from one node to another node in a weighted and directed graph. Each node has an associated weight or cost and the optimal path is the path for which the sum of the costs is minimal. An example is shown in Figure 7-1.

\begin{figure}[h]
\centering
\begin{tabular}{cccc}
9 & 5 & 0 & 9 \\
8 & 7 & 9 & 2 \\
5 & 2 & 5 & 8 \\
7 & 8 & 4 & 3 \\
3 & 5 & 8 & 6 \\
8 & 3 & 2 & 6 \\
9 & 6 & 8 & 7 \\
\end{tabular}\hspace{1cm}
\begin{tabular}{cccc}
9 & 5 & 0 & 9 \\
13 & 7 & 9 & 2 \\
12 & 9 & 7 & 10 \\
16 & 15 & 11 & 7 \\
18 & 16 & g & 13 \\
24 & 18 & 15 & 16 \\
27 & 21 & 23 & 20 \\
\end{tabular}
\caption{Example of dynamic programming. In the cost matrix on the left, each element represents a node with a given cost, which is derived from an image-related cost function. Using back-propagation, an optimal path is sought from the bottom row to the top row. In the matrix on the right side, the minimal cumulative costs are shown and the optimal path (elements in gray) connects minimal cumulative costs per row, resulting in a globally optimal path.}
\end{figure}
Dynamic programming (DP) has been used widely in medical image segmentation, but mainly to enforce spatial continuity. For instance, DP has been widely applied in x-ray left ventricular angiography and coronary angiography. Xu et al. have used DP for automated lung nodule segmentation in computed tomography images. Optimal 2D contours are extracted from all slices using dynamic programming with shape-based constraints. These 2D contours are stacked to obtain 3D surfaces of the nodules. Yamada et al. have presented a 2D dynamic programming-based matching method for kidney glomerulus recognition. In this approach, a parameterization of the shape of the glomerulus is made and constraints on the range of parameters are imposed using dynamic programming. Amini et al. and Geiger et al. have used dynamic programming to find globally optimal solutions to variational problems in energy minimization and to allow direct and natural enforcement of constraints on deformable models.

The approaches described here were all used in static images and typically search for a one-dimensional path through a 2D graph. Little work has been described toward extending DP to higher dimensions. Sonka et al. have proposed an extension of 2D dynamic programming by constructing a cost cube from 2 perpendicular edge images, one for each coronary edge. In this graph, each node corresponds to a combination of possible positions of the left and right coronary borders simultaneously. The optimal path through this graph results in a segmentation, in which shape changes in both coronary edges are coupled. Thedens et al. have developed a graph-searching method for finding the optimal surface through a 3D cost cube. Their approach is based on a data transformation of a 3D lattice into an intermediate 2D graph enabling application of traditional graph-searching techniques. To make this approach computationally feasible, they introduce a heuristic search approach potentially yielding suboptimal solutions. Li et al. propose a computationally feasible solution for finding a surface through a 3D graph lattice by computing a minimum s-t cut through a 3D-directed graph.

7.2.2 Multidimensional Parametric Dynamic Programming

As mentioned before, higher dimensional extensions of dynamic programming have been described for finding either a surface through a 3D volume or finding a path through a 3D volume. In this work, we extend the classic dynamic programming method to higher dimensions, similar to Sonka's approach. An overview of the general method is represented in Figure 7-2, whereas details on the concrete implementation of two tracking problems are described in the next section. Instead of applying dynamic programming directly to image data, we first define a parametric shape
model, expressing the shape with N parameters. An N-dimensional space can be constructed, in which the axes are spanned by the N parameters. A given shape is represented by a point in this parameter space. Thus, a given combination of the N parameters represents a shape instantiation. By discretizing each parameter axis in a limited parameter range, an N-dimensional hypercube is constructed, which is the N-dimensional equivalent of a cost matrix such as shown in Figure 7-1. For each frame in a time sequence of images, such a cost hypercube is calculated.

![Diagram](image)

**Figure 7-2.** General outline of the N-dimensional dynamic programming.
For all possible combinations of the model parameters, i.e., all voxels in the hypercubes, cost values are calculated using a cost function based on image features that express the “goodness of fit” of the model instantiation to the image data. This can be, for instance, the cumulative image gradient along a contour or cross-correlation values between two regions of interest: the choice of the most suitable cost function depends on the application and the selected parameterization. Constraints on temporal or spatial continuity are imposed by setting limits to the allowed parameter changes between subsequent hypercubes for each axis. Subsequently, the optimal path is sought using dynamic programming by connecting the nodes with minimal cost from all hypercubes with each other within these connectivity constraints. This path is a curve in N-dimensional space. The nodes with minimal cost from cost cubes corresponding to different timeframes are connected, in which a connective path through a sequence of hypercubes represents a changing shape in a time sequence (see Figure 7-2). This way, depending on the type of parameterization, spatial and/or temporal continuity can be enforced to ensure smooth motion between frames. The concrete implementation of the shape parameterization depends on the intended application. In the next section, 2 types of applications with corresponding parameterizations are elaborated.

### 7.2.3 Validation Studies

To investigate the performance and accuracy of the proposed method, 2 studies were conducted. The first experiment was performed to investigate the accuracy of the DP method in segmentation of short-axis MR images of the cardiac left ventricle and the effect of imposing time-continuity constraints on the robustness of the segmentation results. The second experiment involves a case study to demonstrate the method’s efficacy in a higher dimensional case: tracking of the aorta in full-cycle MR images using a 6-dimensional parametric representation.

### 7.2.4 Data Material

Cardiac MR imaging examinations were performed in 18 cardiac patients and 2 healthy volunteers. The patients experienced several pathologies, including heart failure (n=8), hypertrophic cardiomyopathy (n=4), transplant follow up (n=3), chest pain or angina (n=3).

Short-axis images of the cardiac left ventricle were scanned using the TrueFISP protocol on a 1.5 T Siemens MR system (Sonata; Siemens Medical Systems, Erlangen, Germany) with a resolution of 256 pixels and a field of view ranging from 340 to 420 mm, which resulted in a pixel size varying
between 1.33 and 1.64 mm. The inter-slice gap was 2 mm, the slice thickness was 8 mm, and the following acquisition parameters were used: TR=3.1 ms, TE=1.6 ms, flip angle 55°, and receiver bandwidth 930 Hz/pixel. In these short-axis MR acquisitions, epi-and endocardial contours were drawn by experts in all slices and all phases following the conventions put forward in the study by Danilouchkine et al.\textsuperscript{23}.

To investigate the applicability of the method to higher dimensional tracking problems, an additional velocity-encoded aorta flow scan was acquired in a patient with congenital cardiac abnormalities. Images were acquired in an image plane perpendicular to the ascending aorta on a Philips Gyroscan 1.5 T MR system using a phase-contrast sequence with a VENC of 3 m/s, field of view 300×253 mm, scan matrix 128×108 reconstructed to 256×256 pixels, pixel size 1.17×1.17 mm (reconstructed), and a slice thickness of 8 mm. The full cardiac cycle was imaged in 30 phases, with TR 14 ms, TE 5.2 ms, and flip angle 20° with 2 signal averages.

7.2.5 Short-Axis Cardiac Magnetic Resonance Segmentation

An elaborate quantitative evaluation was performed in a study on full-cycle contour detection in short-axis cardiac MR images. During the cardiac cycle, the endo-and epicardial borders undergo small deformations from frame to frame; this makes the proposed dynamic programming very suitable for imposing constraints on the maximally allowed deformations between frames of the border positions, possibly yielding better segmentation results.

Using the proposed dynamic programming approach, the optimal contour set for the cardiac cycle can be found as follows (see the flow chart in Figure 7-3 for an overview). After initializing manually by drawing a contour in one phase, the contour is resampled to 32 equiangularly sampled landmarks. Each landmark is parameterized by its coordinates (x,y). (Because the papillary muscles were not included in the manual drawing conform clinical standards, 32 landmarks provided sufficient detail to accurately describe the approximately circular contour shapes.). Each landmark is tracked separately over time by defining a mask in each landmark and a search space around each landmark by setting limits to the displacement parameters dx and dy, which are the allowed shifts for the center of the mask in x and y directions. The mask is positioned onto all allowed positions in the defined search space, and a cross correlation coefficient is calculated for each position with the mask in the reference frame.
Figure 7-3. Overview of cardiac left ventricle segmentation using multidimensional dynamic programming.

Subsequently, for each landmark, the cost hypercubes are filled with the cost values associated with all shifts of the center of the mask. A 3D search space is created by combining these cost matrices, obtaining a 3D graph. The first and last frames of the graph are the matrices corresponding to the frame with the manually annotated landmarks, i.e., the initialization frame. For the cost matrix corresponding to the landmarks in the initialization
frame, the center value is set to zero and all other values are set to high values, as to leave this landmark position unchanged. To find the optimal solution, a connective path is sought through the 3D graph, starting at the zero element in the first slice and crossing each layer of the graph in only one point. Each path through the graph is a possible solution. The total cost of a path is the sum of the costs of all the nodes constituting the path. The optimal solution is the path with minimal total cost. This optimal path yields translation vectors for each landmark with respect to the manually annotated landmark in the reference frame, thus giving the optimal positions for all landmarks over the full cycle.

While searching the optimal path through the graph, hard constraints can be imposed, e.g. a continuity constraint allowing a maximum side step. Because the sampling rate of the full cycle images is sufficiently high, the inter-frame movement of the borders does not exceed 2 pixels. Therefore, a maximal side step of 2 pixels was allowed to obtain time continuity between frames. To assess the effect of the constraints on the side-step parameter, experiments were repeated without side-step constraints. Initialization was performed manually in a mid-systolic phase to enable automatic detection of both end-diastolic and end-systolic frames. This way, an automatic calculation of ejection fraction was possible without introducing a bias toward either the ED or ES frame. In all slices between apex and base, endo-and epicardial contours were available for the initialization frame.

**Figure 7-4.** Possible settings for the shape of the mask: ellipsoid perpendicular to contour (−½), circular mask (0), and ellipsoid tangent to contour (+½).

### 7.2.6 Parameter Selection

Several parameters influence the performance of the method, and a brute-force search was performed to systematically select a parameter combination yielding good segmentation results. The parameters involved in the brute force search were:
• Type of cost function: 2 types of cost functions were considered. First, cross correlation of the mask in the current frame with the mask in the reference frame was considered. The cross correlation coefficients were considered as costs, i.e. the higher the correlations, the lower the cost. Second, the sum of absolute gray value differences between the masks was used. Here, a smaller difference corresponds to a lower cost value.

• Mask size: The mask size and shape define the region of interest around each landmark and the amount of information involved in the calculation of the costs. The mask must contain sufficient structure for the cost values to be accurate. However, a too-large mask may include papillary muscles or other structures, which are not present in every frame, potentially resulting in wrong landmark positions in some frames. In the detection of endocardial contours, mask clipping was applied. Because in apical slices the endocardial contours may become very small, the mask size is adapted to half of the radius of the endocardial contour in the initialization frame.

• Mask shape: The shape of the mask is varied from an ellipse tangent to the contour to a circular shape to an ellipse perpendicular to the contour by changing the parameter value between $-\frac{1}{2}$ and $+\frac{1}{2}$ (see Figure 7-4).

• Physical dimensions of the search space: The parameter limits of the search space were defined by the maximally allowed displacements in x- and y-directions. The size of the search space should be large enough to follow the movement of the myocardium between end-diastolic and end-systolic phases.

The first parameter involves the selection of cost function and is the only intensity feature-dependent choice. The other parameters describe geometric and kinematic properties of the contracting heart. Therefore, we expect the parameter selection to be largely independent of the scanning protocol, therefore generalizing well toward other cardiac MR acquisition protocols not tested here.

**7.2.7 Evaluation Indices**

To quantify the matching accuracy of the model, the automatically detected contours were compared with the manually defined expert contours on the basis of the point-to-curve border positioning errors, the ED and ES volumes and the EF. The point-to-curve errors were defined as the shortest distance between an automatically detected landmark and the manually drawn contour, i.e. the distance along the normal to the manually drawn curve. Average and maximum border positioning errors were measured. The ED and ES volumes were calculated as follows. In each slice, the
surface of the endocardial border, i.e. the blood pool was calculated and multiplied with the slice thickness. The sum of all slice volumes was calculated to determine the 3D volume of the cardiac left ventricle. Using the volumes in ED and ES phases of the cardiac cycle, the EF was calculated as follows:

$$EF = \frac{V_{ED} - V_{ES}}{V_{ED}}$$  \hspace{1cm} (7.1)$$

where EF is the ejection fraction, $V_{ED}$ the cardiac left ventricular volume in end-diastolic phase, and $V_{ES}$ the volume in end-systolic phase.

Paired-sample t-tests were performed to determine if the errors in border positioning, volumes, and ejection fractions were significant. P values smaller than 0.05 were considered significant.

**Figure 7-5.** Six-dimensional parameterization of a region around the aorta with center coordinates $(x, y)$ and 4 radii $r_1, r_2, r_3, r_4$.

### 7.2.8 Aorta Tracking

An additional experiment was performed to investigate the potential of multidimensional dynamic programming applied to higher dimensional cases. The area of the aorta was tracked over time to measure the flow in images of a specific slice of the aorta, which are used to measure flow volume. In a time sequence of images, translation and deformation of the aorta occurs. To this end, the shape of the aorta was parameterized with 6 parameters (center coordinates $[x, y]$ and 4 radii $r_1$–$r_4$) as shown in Figure 7-5. In sequential images, the center coordinates were varied as well as the 4 radii. A spline was fitted to the end points of the radii to obtain an ellipse. With this parameterization, a 6-dimensional cost hypercube was constructed. The costs associated with each instantiation of the aorta shape were defined as the cumulative image gradient along the model contour. Ranges were defined for all combinations of the mentioned parameters, i.e,
all shape instantiations. The associated costs were calculated and stored in the corresponding positions in the hypercube. Next, using conventional backtracking, an optimal path with minimal total cost within time-continuity constraints was sought through the hypercubes. The optimal path yielded translation vectors for the center coordinates and deformation vectors for the radii with respect to the reference frame. The results of the experiments were evaluated visually.

<table>
<thead>
<tr>
<th>Table 7-1. Optimal settings for parameters obtained with a brute-force search *</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contour</td>
</tr>
<tr>
<td>Endocardial</td>
</tr>
<tr>
<td>Epicardial</td>
</tr>
</tbody>
</table>

* Mask shape 0 corresponds to a circular mask (see Figure 7-3) and -½ corresponds to ellipsoid. Results for endocardial and epicardial contour are shown.

7.3 RESULTS

7.3.1 Parameter Settings
The parameters computed in the parameter selection process are given in Table 7-1. For detection of both endocardial and epicardial contours, the cross correlation-based cost function was found to give more accurate results than the cost function based on the sum of absolute differences.

7.3.2 Segmentation Results
Epicardial and endocardial contours were detected separately. With parameter settings as shown in Table 7-1, the average border positioning error (BPE) over all slices, all phases, and all studies was 1.77±0.57 mm (average maximum BPE was 5.21 mm) for epicardial contours and 1.86±0.59 mm (average maximum BPE was 4.51 mm) for endocardial contours. In Table 7-2, the errors in volumes and ejection fractions are shown. Figures 7-6 shows Bland-Altman plots for ED and ES volumes and ejection fractions. Parameters of the regression fits are given in Table 7-3.

Figure 7-7 shows that if no time-continuity constraints are imposed, the segmentations deteriorate and large inter-frame discontinuities and border positioning errors are introduced. As can be seen from the bottom row of Figure 7-7, single landmarks can make large excursions and lock onto false positions, resulting in bad segmentation results.
Table 7-2. Average manual ED and ES volumes and ejection fractions, with average errors made by an automatic dynamic programming-based method

<table>
<thead>
<tr>
<th></th>
<th>Average manual</th>
<th>Average error</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>ED volume (ml)</td>
<td>171.68 ± 91.10</td>
<td>4.24 ± 4.62</td>
<td>6.14 E-4</td>
</tr>
<tr>
<td>ES volume (ml)</td>
<td>103.96 ± 95.96</td>
<td>-4.36 ± 4.26</td>
<td>2.05 E-4</td>
</tr>
<tr>
<td>Ejection fraction (%)</td>
<td>47.73 ± 19.20</td>
<td>4.82 ± 3.01</td>
<td>8.48 E-7</td>
</tr>
</tbody>
</table>

Typical computation times for full-cycle segmentation of a 3D short-axis cardiac MR scan (8-12 slices, 20-25 frames) amounted to approximately 4 minutes on a desktop PC with an AMD Athlon 1.8 GHz processor.

Figure 7-6. Bland-Altman plot for end-diastolic volumes (top left), end-systolic volumes (top right) and ejection fraction (bottom).
7.3.3 Tracking Results

Figure 7-8 shows the result of tracking the aorta in 30 sequential timeframes using a shape parameterization with 4 radii and 2 center coordinates.

<table>
<thead>
<tr>
<th></th>
<th>Regression fit</th>
<th>$R^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ED volumes</td>
<td>$y = 0.99x - 2.21$</td>
<td>0.998</td>
</tr>
<tr>
<td>ES volumes</td>
<td>$y = 0.98x - 6.55$</td>
<td>0.998</td>
</tr>
<tr>
<td>Ejection fractions</td>
<td>$y = 0.89x - 0.34$</td>
<td>0.984</td>
</tr>
</tbody>
</table>

7.4 DISCUSSION

A dynamic programming-based method for time-continuous segmentation of endo-and epicardial contours was presented. Conventional 2D dynamic programming was extended to higher dimensions and applied to 2 substantially different tracking and segmentation problems to illustrate the method’s performance in multiple dimensions.

In the quantitative evaluation, segmentations were performed in 20 studies in all phases and all slices. Contour detection was successful in all included imaging slices and the average BPE was very good (1.86 mm and 1.77 mm for endo-and epicardial BPE, which corresponds to approximately 1 pixel), although slightly less accurate as interobserver BPE reported in the study by van Assen et al.\(^1\) (1.27 mm and 1.14 mm for endo-and epicardial BPE, respectively). Lötjönen et al.\(^2\) report an average segmentation error of 2.57 mm using the probabilistic atlas based method. The method presented by Kaus et al.\(^3\) has a mean deviation from manual segmentations of 2.45 mm in end-diastolic phase and 2.84 mm in end-systolic phase, whereas van Assen et al. report 2.24 mm and 2.84 mm errors for endo-and epicardium, respectively. Lorenzo-Valdés et al.\(^8\) report an average segmentation error of 2.21 mm for the 3 middle slices of the left ventricle over all timeframes. Also, the maximum BPE for endo-and epicardial contours (4.51 mm and 5.21 mm, respectively) compares highly favorably to the other automated methods (11–15 pixels as reported in the studies by van Assen et al.\(^1\) and Kaus et al.\(^3\)) and compares well to interobserver maximum errors reported in the study by van Assen et al.\(^1\) (4.34 and 3.93 maximum BPE for endo-and epicardial contours, respectively). Therefore, we can conclude that border positioning errors presented in this article compare favorable to these other results reported in recent literature and approach interobserver variabilities resulting from manual contour drawing. Of these methods, only Lorenzo-Valdes reports
full-cycle contour detection. The other approaches report results in ED and ES phases only. An important criterion for any automated border detection method for cardiac MR images is that if morphology is important, accurate regional measurements require the border detection to yield errors at least as good as the interobserver variability resulting from manual contour drawing, because a small error can already have a significant impact on regional measurements such as wall thickness and wall thickening. Of the algorithms compared here, the proposed method best approaches this interobserver BPE.

Figure 7-7. The top row shows segmentations obtained with temporal continuity constraint in 5 successive frames. The first frame is the mid-systolic phase which was used for manual initialization. In the bottom row, the same frames segmented without imposing temporal continuity constraints are shown.

The automatically determined end-diastolic and end-systolic volumes showed near-perfect correlations with the volumes derived from manual contours. Also, the ejection fractions of automatically segmented and manually drawn studies correlated very well (see Table 7-3). This is a direct consequence of the small border-positioning errors, but although the volume errors are small, they are statistically significant (P<0.05). Bland-Altman plots reveal a slight systematic underestimation of the ED volumes and a slight systematic overestimation of the ES volumes. This, of course, has a negative influence on the ejection fraction estimates. The systematic over-and underestimations are most probably caused by a bias toward the reference frame. Currently, a rectangular search space around the landmarks is used (dx, dy). Defining a different search space shape, eg, allowing more radial movement, might resolve the systematic bias. The errors, however, are small and not clinically relevant; they are in the same range as interobserver variations in ventricular volumes and ejection fractions\textsuperscript{23}, comparing measurements from manually drawn contours from different observers.
As can be seen in Table 7-1, a larger search space and mask need to be defined for endocardial contour detection than for epicardial contour detection. Also, the border positioning errors are slightly larger for endocardial contours. This is caused by the fact that during the cardiac cycle, the movement of the endocardial border has larger amplitude than the epicardial border. Therefore, the texture in the masks used for tracking the endocardial borders demonstrates relatively more changes between end-diastolic and end-systolic phases. Also, the fusion of the papillary muscles with the myocardium in end-systolic phases substantially changes the mask texture. These effects deteriorate the cross correlation values, i.e. the cost function outcomes.

![Figure 7-8](image)

Figure 7-8. Sequential timeframes in an aortic flow scan, in which the motion and deformation aorta is tracked over the cardiac cycle. Note the smooth transitions between frames, which is a result of the imposed time continuity.

The proposed method may possibly be further improved, eg, by blurring the images before contour detection. Blurring the images reduces noise and better cross correlation values can be obtained. Another improvement might be smoothing of the contours after detection. Currently, all landmarks are detected independently and are connected with each other by straight lines. This in general gives a slight underestimation of the volumes. The underestimation is larger in basal slices and also in
end-diastolic images, because the sampling of the contours is less dense in these images. Smoothing the contours after connecting the landmarks might give a better estimate of the volumes, although landmark positions remain unchanged.

Experiments have shown that adding a time constraint to the conventional dynamic programming approach substantially improved the segmentation results. Without this constraint, single landmarks showed large inter-frame displacements, which led to locking on false edges. Imposing a maximal inter-frame displacement by means of a time constraint resolved these errors. In the current approach, endo-and epicardial contours are detected separately. However, the locations of both contours are strongly correlated. Therefore, combining both contours and detecting them simultaneously might yield still more robust results.

For each landmark, separate cost matrices are calculated for each timeframe, and the optimal positions of the landmarks are determined independently from its neighboring landmarks. In some cases, this leads to crossing landmarks and spatial discontinuities in the segmentations (see Figure 7-9). These artifacts may be resolved by following the approach in the study by Thedens et al.\textsuperscript{20} and coupling neighboring landmarks to each other, preventing landmarks from crossing each other. This way, not only time-continuous, but also spatially continuous segmentations can be obtained, which may also lead to a further improvement of the border positioning errors, the volume estimations, and ejection fraction calculations.

![Figure 7-9. Spatial discontinuity caused by crossing landmarks in endocardial contour.](image)
In our current approach, segmentations were performed on a slice-per-slice basis. Time-continuous segmentation results can be obtained for each slice in this manner. However, inter-slice continuity is not guaranteed. By coupling landmarks from neighboring slices with the proposed approach, inter-slice continuity may be obtained as well.

The parameter selection procedure involved systematically deriving a good parameter combination, in which most of the parameters were related to geometry and cardiac kinematics, which are independent of the applied scanning protocol. Highest accuracy was obtained with the presented values for the parameters in the tested image sets. However, we expect similar performance in other MR datasets and protocols as a result of the protocol independence of the tuned parameters.

The aorta tracking experiments have demonstrated the power of the proposed method in a 6-dimensional case study. The proposed approach was used for automatic tracking of the aorta in a time sequence of images, and visual inspection showed that a time-continuous result was obtained. This example showed that there are no theoretical obstacles for the expansion of the proposed method to high dimensions; however, further evaluation is required to quantify the performance of dynamic programming in such higher-dimensional cases.

The main limitation of the proposed method is its computational complexity, which increases exponentially with the number of parameter dimensions. Finding the optimal path through cost hypercubes is fast; however, filling the cost hypercubes may become time-consuming. This obviously depends on the dimension of the hypercubes and the type of cost function, e.g. sum of absolute differences, cross-correlation values, and so on. In the experiments that were performed, typical computing times in the order of 3 to 5 minutes per study (on average, 20 phases and 10 slices) were found. Because the analysis is performed offline and with increasing computer power, this does not impose a major limitation on clinical applicability. In addition, by pruning the search space, substantial additional performance gains can be achieved.

7.5 Conclusion

In conclusion, the proposed method has shown great potential in tracking and segmentation of cardiac MR time sequences. The multidimensional dynamic programming allows for direct and natural enforcement of constraints, e.g. shape-based constraints or time-continuity constraints. It is not iterative and therefore it is exact and stable. With dynamic programming, global optimality of the solution is ensured and multidimensional dynamic programming enables detection of time evolution
and full-cycle time-continuous segmentation in a series of images. The reported results compare favorably to the best-reported results in recent literature, underlining the potential of this method for application in daily clinical practice.

7.6 References


